

dsPIC33FJXXXGPX06/X08/X10

数据手册

高性能 16 位
数字信号控制器

© 2007 Microchip Technology Inc.

请注意以下有关 Microchip 器件代码保护功能的要点：

- Microchip 的产品均达到 Microchip 数据手册中所述的技术指标。
- Microchip 确信：在正常使用的情况下， Microchip 系列产品是当今市场上同类产品中最安全的产品之一。
- 目前，仍存在着恶意、甚至是非法破坏代码保护功能的行为。就我们所知，所有这些行为都不是以 Microchip 数据手册中规定的操作规范来使用 Microchip 产品的。这样做的人极可能侵犯了知识产权。
- Microchip 愿与那些注重代码完整性的客户合作。
- Microchip 或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是 “牢不可破”的。

代码保护功能处于持续发展中。 Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了《数字器件千年版权法案 （Digital Millennium Copyright Act）》。如果这种行为导致他人在未经授权的情况下，能访问您的软件或其他受版权保护的成果，您有权依据该法案提起诉讼，从而制止这种行为。

提供本文档的中文版本仅为了便于理解。请勿忽视文档中包含的英文部分，因为其中提供了有关 Microchip 产品性能和使用情况的有用信息。Microchip Technology Inc. 及其分公司和相关公司、各级主管与员工及事务代理机构对译文中可能存在的任何差错不承担任何责任。建议参考 Microchip Technology Inc. 的英文原版文档。

本出版物中所述的器件应用信息及其他类似内容仅为您提供便利，它们可能由更新之信息所替代。确保应用符合技术规范，是您自身应负的责任。**Microchip 对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保，包括但不限于针对其使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。** Microchip 对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将 Microchip 器件用于生命维持和 / 或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时，会维护和保障 Microchip 免于承担法律责任，并加以赔偿。在 Microchip 知识产权保护下，不得暗中或以其他方式转让任何许可证。

商标

Microchip 的名称和徽标组合、 Microchip 徽标、 Accuron、 dsPIC、 KEELOQ、 KEELOQ 徽标、 microID、 MPLAB、 PIC、 PICmicro、 PICSTART、 PRO MATE、 rfPIC 和 SmartShunt 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

AmpLab、 FilterLab、 Linear Active Thermistor、 Migratable Memory、 MXDEV、 MXLAB、 SEEVAL、 SmartSensor 和 The Embedded Control Solutions Company 均为 Microchip Technology Inc. 在美国的注册商标。

Analog-for-the-Digital Age、 Application Maestro、 CodeGuard、 dsPICDEM、 dsPICDEM.net、 dsPICworks、 dsSPEAK、 ECAN、 ECONOMONITOR、 FanSense、 FlexROM、 fuzzyLAB、 In-Circuit Serial Programming、 ICSP、 ICEPIC、 Mindi、 MiWi、 MPASM、 MPLAB Certified 徽标、 MPLIB、 MPLINK、 PICkit、 PICDEM、 PICDEM.net、 PICLAB、 PICtail、 PowerCal、 PowerInfo、 PowerMate、 PowerTool、 REAL ICE、 rfLAB、 Select Mode、 Smart Serial、 SmartTel、 Total Endurance、 UNI/O、 WiperLock 和 ZENA 均为 Microchip Technology Inc. 在美国和其他国家或地区的商标。

SQTP 是 Microchip Technology Inc. 在美国的服务标记。

在此提及的所有其他商标均为各持有公司所有。

© 2007, Microchip Technology Inc. 版权所有。

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip 位于美国亚利桑那州 Chandler 和 Tempe 与位于俄勒冈州 Gresham 的全球总部、设计和晶圆生产厂及位于美国加利福尼亚州和印度的设计中心均通过了 ISO/TS-16949:2002 认证。公司在 PIC® MCU 与 dsPIC® DSC、KEELOQ® 跳码器件、串行 EEPROM、单片机外设、非易失性存储器和模拟产品方面的质量体系流程均符合 ISO/TS-16949:2002。此外，Microchip 在开发系统的设计和生产方面的质量体系也已通过了 ISO 9001:2000 认证。

© 2007 Microchip Technology Inc.

MICROCHIP dsPIC33FJXXXGPX06/X08/X10

高性能 16 位数字信号控制器

工作范围：

- DC – 40 MIPS（3.0-3.6V、-40°C 至 +85°C 时为 40 MIPS）
- 工业级温度范围 （-40°C 至 +85°C）

高性能 DSC CPU：

- 改进型哈佛架构
- C 编译器优化的指令集
- 16 位宽数据总线
- 24 位宽指令
- 可寻址最大 4M 指令字的线性程序存储空间
- 可寻址最大 64 KB 的线性数据存储空间
- 83 条基本指令：多为单字 / 单周期指令
- 16 个 16 位通用寄存器
- 两个 40 位累加器：
  - 带舍入和饱和选择
- 灵活和强大的寻址模式：
  - 间接寻址、模寻址和位反转寻址
- 软件堆栈
- 16 x 16 位小数 / 整数乘法运算
- 32/16 位和 16/16 位除法运算
- 单周期乘－累加运算：
  - DSP 运算的累加器回写操作
  - 双数据取操作
- 可将最多 40 位数据左移或右移最多 16 位

直接存储器访问 （DMA）：

- 8 通道硬件 DMA
- 2 KB 双端口 DMA 缓冲区 （DMA RAM），用于存储通过 DMA 传输的数据：
  - 允许在 CPU 执行代码期间在 RAM 和外设间传输数据 （不额外占用周期）
- 大多数外设支持 DMA

中断控制器：

- 中断响应延时为 5 个周期
- 118 个中断向量
- 最多 63 个中断源
- 最多 5 个外部中断
- 7 个可编程优先级
- 5 个处理器异常

数字 I/O：

- 最多 85 个可编程数字 I/O 引脚
- 最多 24 个引脚上具有唤醒 / 电平变化中断功能
- 输出引脚可驱动 3.0V 至 3.6V 的电压
- 所有数字输入引脚可承受 5V 的电压
- 所有 I/O 引脚的最大拉 / 灌电流为 4 mA

片上闪存和 SRAM：

- 闪存程序存储器，最大 256 KB
- 数据 SRAM，最大 30 KB （包括 2 KB 的 DMA RAM）

系统管理：

- 灵活的时钟选择：
  - 外部振荡器、晶振、谐振器和内部 RC 振荡器
  - 全集成 PLL
  - 极低抖动 PLL
- 上电延时定时器
- 振荡器起振定时器 / 稳定器
- 自带 RC 振荡器的看门狗定时器
- 故障保护时钟监视器
- 多个复位源

功耗管理：

- 片上 2.5V 稳压器
- 实时时钟源切换
- 可快速唤醒的空闲、休眠和打盹模式

定时器 / 捕捉 / 比较 /PWM：

- 定时器 / 计数器，最多 9 个 16 位定时器：
  - 最多可以配对作为 4 个 32 位定时器使用
  - 1 个定时器可依靠外部 32.768 kHz 振荡器作为实时时钟使用
  - 可编程预分频器
- 输入捕捉 （最多 8 路通道）：
  - 上升沿捕捉、下降沿捕捉或上升 / 下降沿捕捉
  - 16 位捕捉输入功能
  - 每路捕捉通道都带有 4 字深度的 FIFO 缓冲区
- 输出比较 （最多 8 路通道）：
  - 1 个或 2 个 16 位比较模式
  - 16 位无毛刺 PWM 模式

© 2007 Microchip Technology Inc.

通信模块：

- 3 线 SPI （最多 2 个模块）：
  - 帧支持简单编解码器的 I/O 接口
  - 支持 8 位和 16 位数据
  - 支持所有串行时钟格式和采样模式
- I^2C™ （最多 2 个模块）：
  - 完全支持多主机从模式
  - 7 位和 10 位寻址
  - 总线冲突检测和仲裁
  - 集成信号调理
  - 从地址掩码
- UART （最多 2 个模块）：
  - 检测到地址位时产生中断
  - 出现 UART 错误时产生中断
  - 检测到启动位时将器件从休眠模式唤醒
  - 4 字符深度的发送和接收 FIFO 缓冲区
  - LIN 总线支持
  - 硬件 IrDA® 编解码
  - 高速波特率模式
  - 使用 CTS 和 RTS 的硬件流控制
- 数据转换器接口 （Data Converter Interface，DCI）模块：
  - 编解码器接口
  - 支持 I^2S 和 AC'97 协议
  - 最多 16 位数据字，每帧最多 16 字
  - 4 字深度的发送和接收缓冲区
- 增强型 CAN （ECAN™ 模块） 2.0B active 版本（最多 2 个模块）：
  - 最多 8 个发送缓冲区和 32 个接收缓冲区
  - 16 个接收过滤器和 3 个屏蔽寄存器
  - 用于诊断和总线监视的环回模式、监听模式和监听所有报文模式
  - 收到 CAN 报文时唤醒器件
  - 自动处理远程发送请求
  - 使用 DMA 的 FIFO 模式
  - DeviceNet™ 寻址支持

模数转换器 （Analog-to-Digital Converters， ADC）：

- 一个器件中最多 2 个 ADC 模块
- 10 位 1.1 Msps 或 12 位 500 Ksps 转换：
  - 2、 4 或 8 路同时采样
  - 最多 32 路带自动扫描功能的输入通道
  - 可手动启动转换或与 4 个触发源中的一个同步
  - 休眠模式下仍可进行转换
  - 积分非线性误差最大为 ±1 LSb
  - 微分非线性误差最大为 ±1 LSb

CMOS 闪存技术：

- 低功耗高速闪存技术
- 全静态设计
- 3.3V （±10%）工作电压
- 工业级温度
- 低功耗

封装：

- 100 引脚 TQFP （14x14x1 mm 和 12x12x1 mm）
- 80 引脚 TQFP （12x12x1 mm）
- 64 引脚 TQFP （10x10x1 mm）

注：	关于每个器件的具体外设特性，请参见相应的器件数据表。

© 2007 Microchip Technology Inc.

dsPIC33F 产品系列

dsPIC33F 系列器件具有一种子系列，即通用系列，它是各种 16 位 MCU 嵌入式应用的理想选择。

其中带有编解码器接口的器件非常适合语音和音频处理应用。

下表列出了每个系列的器件名称、引脚数、存储容量和可用的外设，表后还附有它们的引脚图。

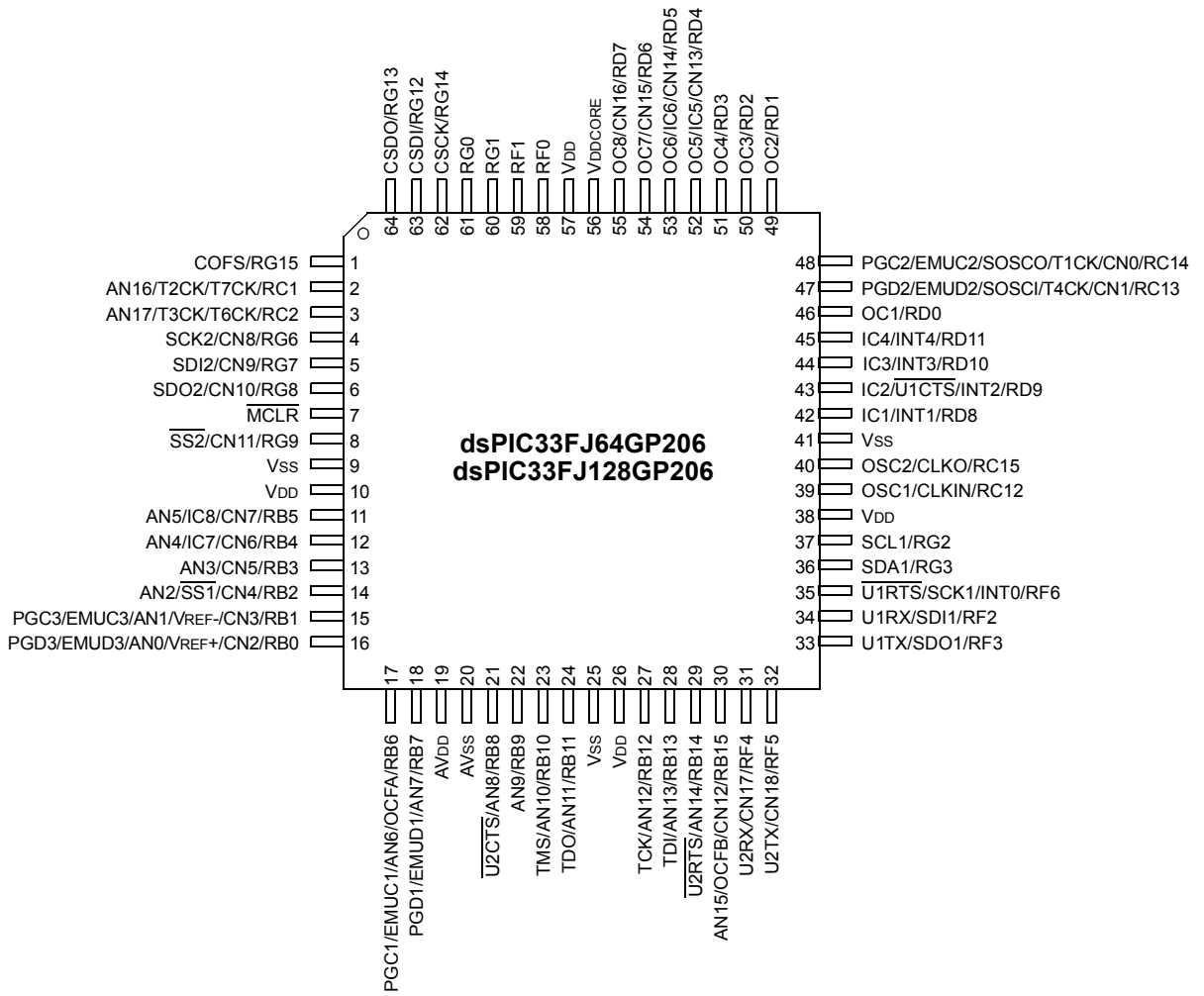
dsPIC33F 通用系列器件

器件	引脚数	闪存程序存储器（KB）	RAM（KB）(1)	16 位定时器	输入捕捉	输出比较标准 PWM	编解码器接口	ADC	UART	SPI	I²C™	增强型 CAN	最大 I/O 引脚数 (2)	封装
dsPIC33FJ64GP206	64	64	8	9	8	8	1	1 个 ADC，18 路通道	2	2	1	0	53	PT
dsPIC33FJ64GP306	64	64	16	9	8	8	1	1 个 ADC，18 路通道	2	2	2	0	53	PT
dsPIC33FJ64GP310	100	64	16	9	8	8	1	1 个 ADC，32 路通道	2	2	2	0	85	PF, PT
dsPIC33FJ64GP706	64	64	16	9	8	8	1	2 个 ADC，18 路通道	2	2	2	2	53	PT
dsPIC33FJ64GP708	80	64	16	9	8	8	1	2 个 ADC，24 路通道	2	2	2	2	69	PT
dsPIC33FJ64GP710	100	64	16	9	8	8	1	2 个 ADC，32 路通道	2	2	2	2	85	PF, PT
dsPIC33FJ128GP206	64	128	8	9	8	8	1	1 个 ADC，18 路通道	2	2	1	0	53	PT
dsPIC33FJ128GP306	64	128	16	9	8	8	1	1 个 ADC，18 路通道	2	2	2	0	53	PT
dsPIC33FJ128GP310	100	128	16	9	8	8	1	1 个 ADC，32 路通道	2	2	2	0	85	PF, PT
dsPIC33FJ128GP706	64	128	16	9	8	8	1	2 个 ADC，18 路通道	2	2	2	2	53	PT
dsPIC33FJ128GP708	80	128	16	9	8	8	1	2 个 ADC，24 路通道	2	2	2	2	69	PT
dsPIC33FJ128GP710	100	128	16	9	8	8	1	2 个 ADC，32 路通道	2	2	2	2	85	PF, PT
dsPIC33FJ256GP506	64	256	16	9	8	8	1	1 个 ADC，18 路通道	2	2	2	1	53	PT
dsPIC33FJ256GP510	100	256	16	9	8	8	1	1 个 ADC，32 路通道	2	2	2	1	85	PF, PT
dsPIC33FJ256GP710	100	256	30	9	8	8	1	2 个 ADC，32 路通道	2	2	2	2	85	PF, PT

注 **1：** RAM 中包括 2 KB 的 DMA RAM。
2： 最大 I/O 引脚数包括与外设功能复用的引脚。

引脚图

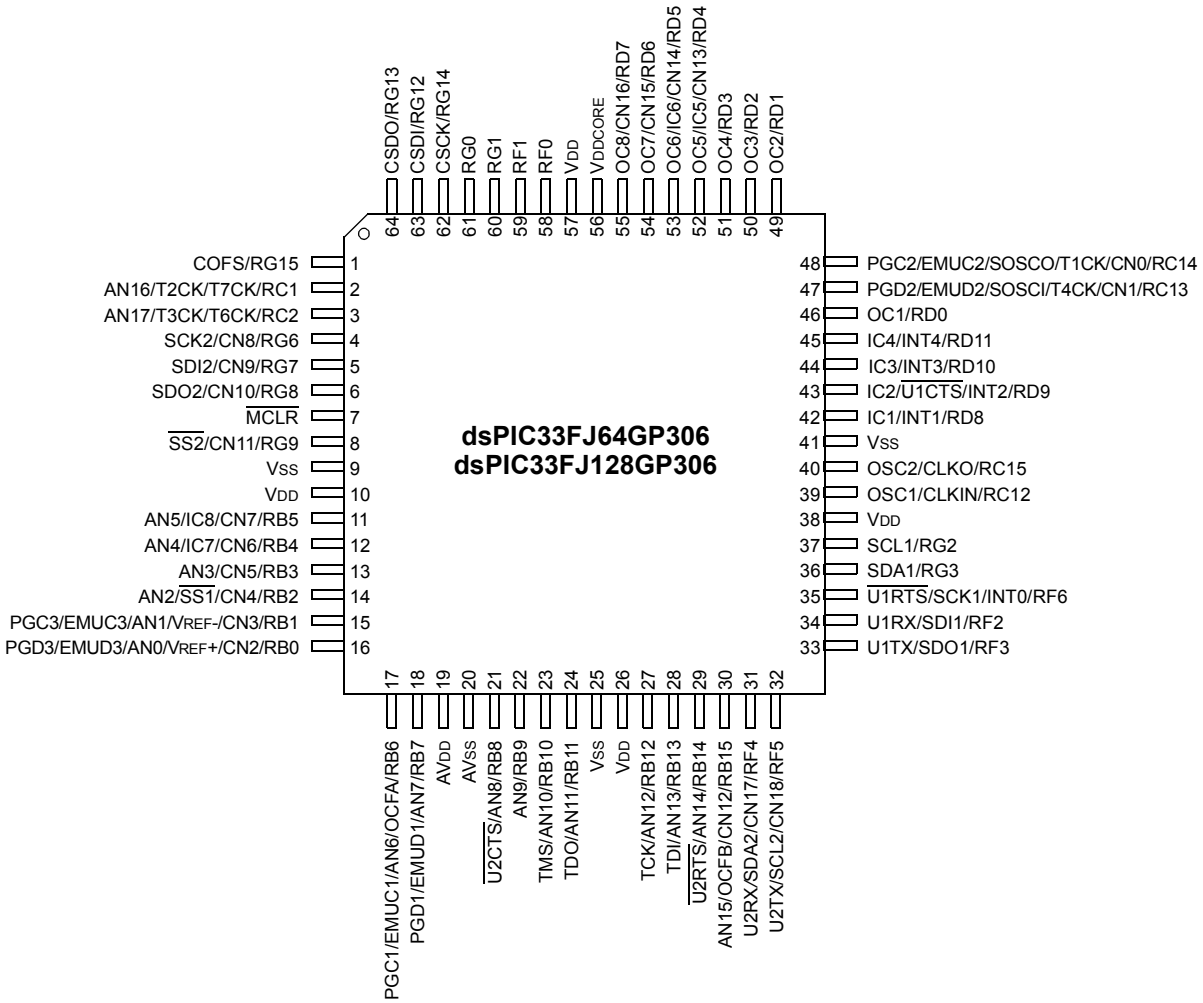
64 引脚 TQFP



© 2007 Microchip Technology Inc.

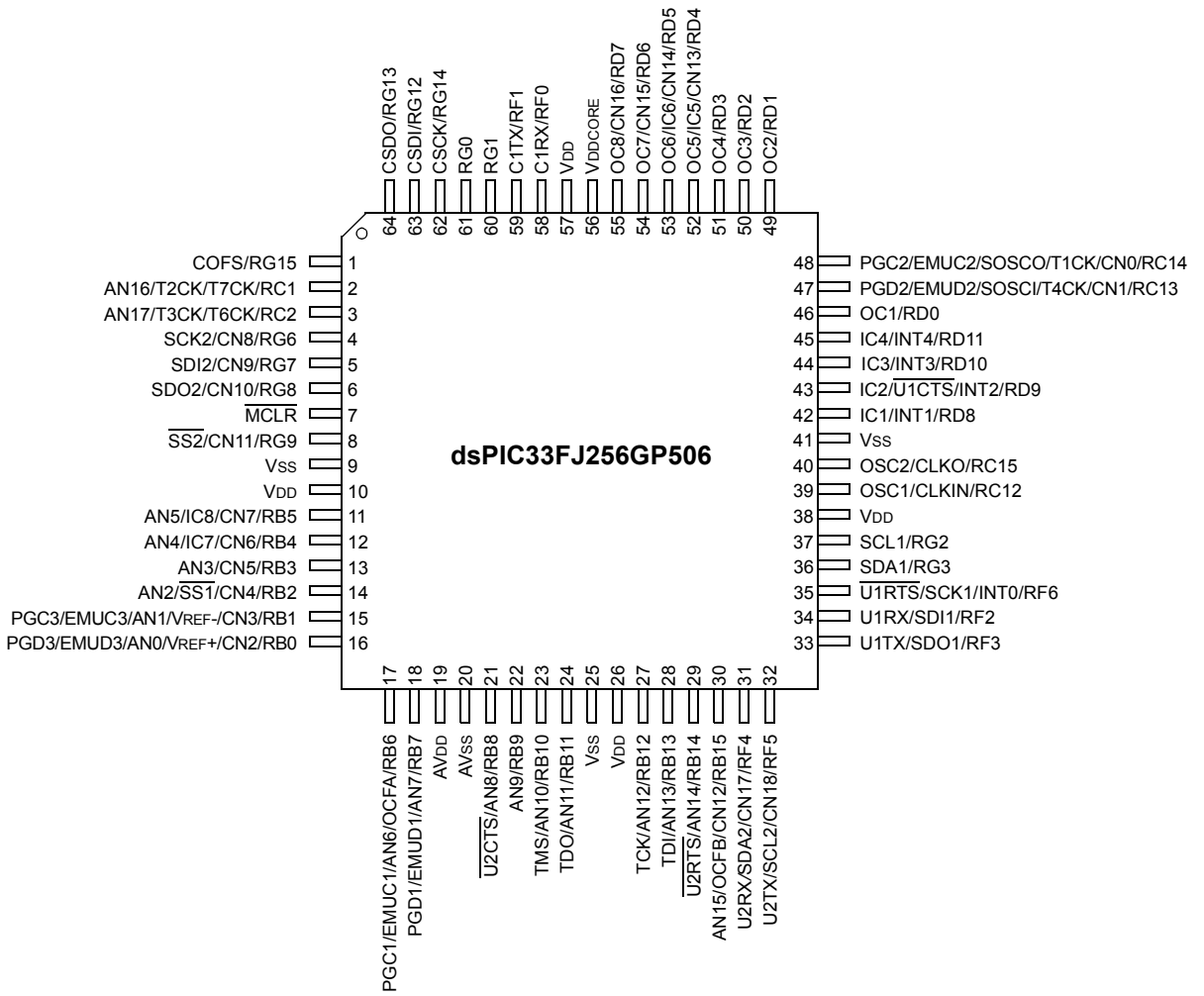
引脚图 （续）

64 引脚 TQFP



引脚图 （续）

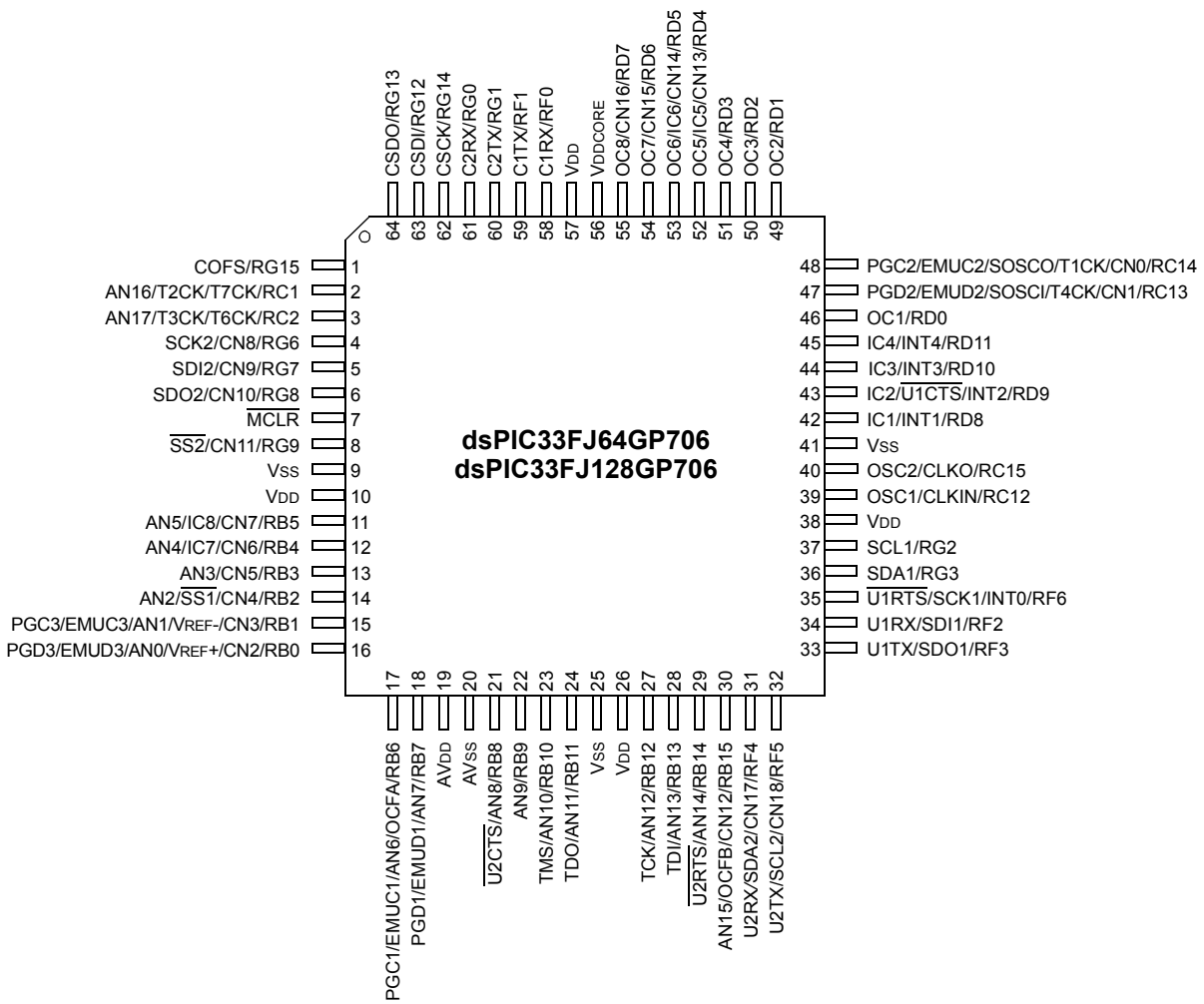
64 引脚 TQFP



© 2007 Microchip Technology Inc.

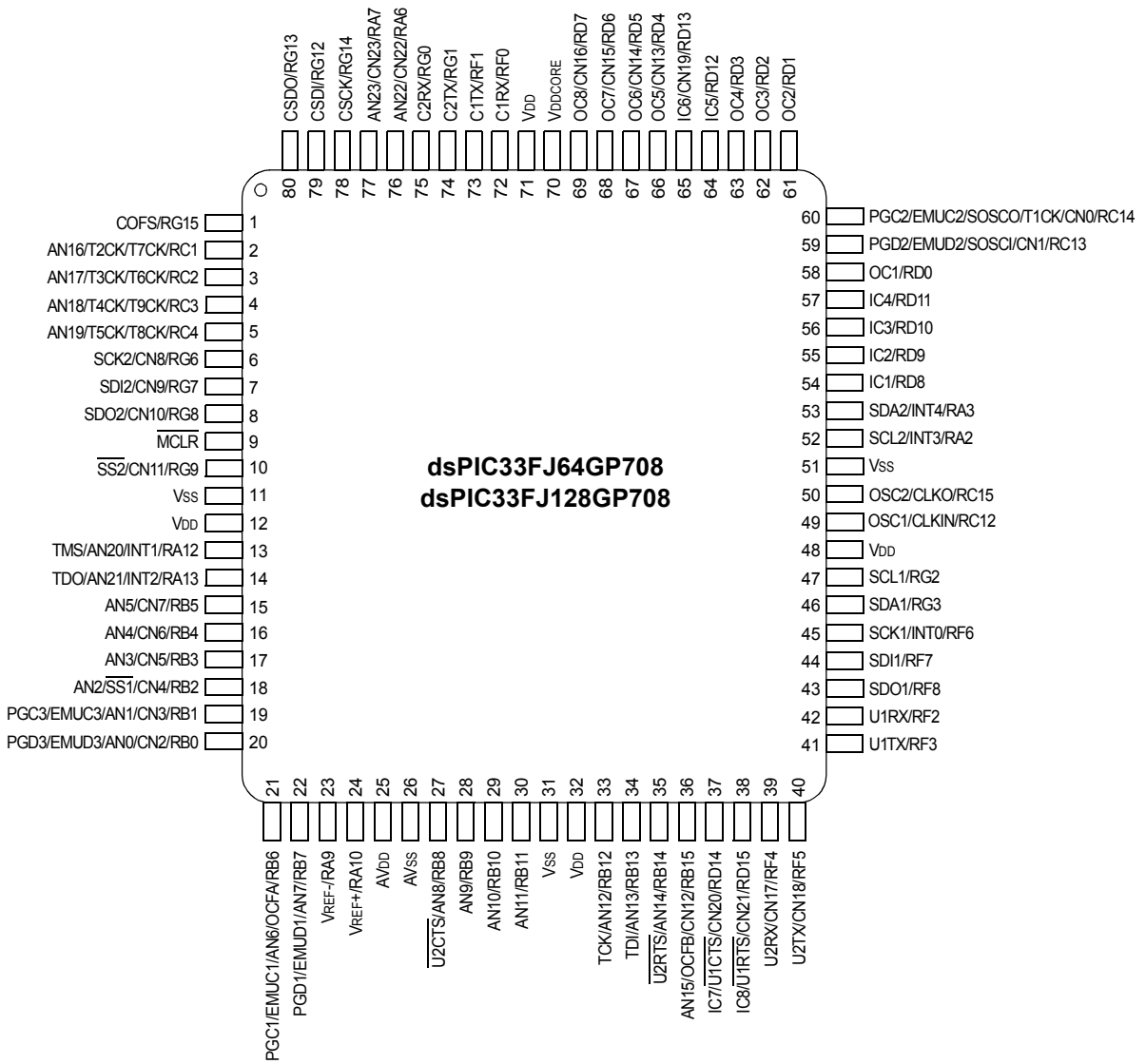
引脚图 （续）

64 引脚 TQFP



引脚图 （续）

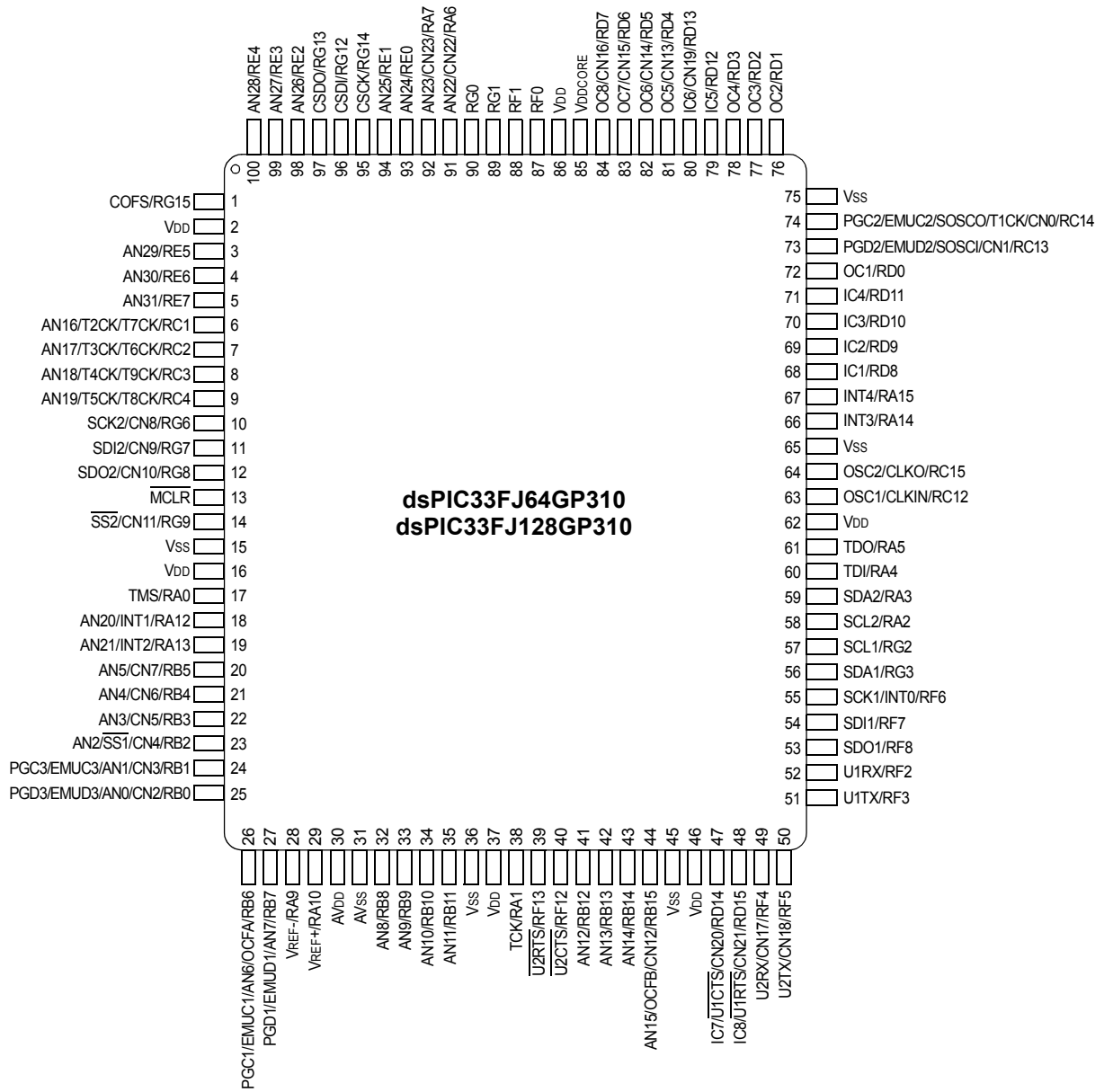
80 引脚 TQFP



© 2007 Microchip Technology Inc.

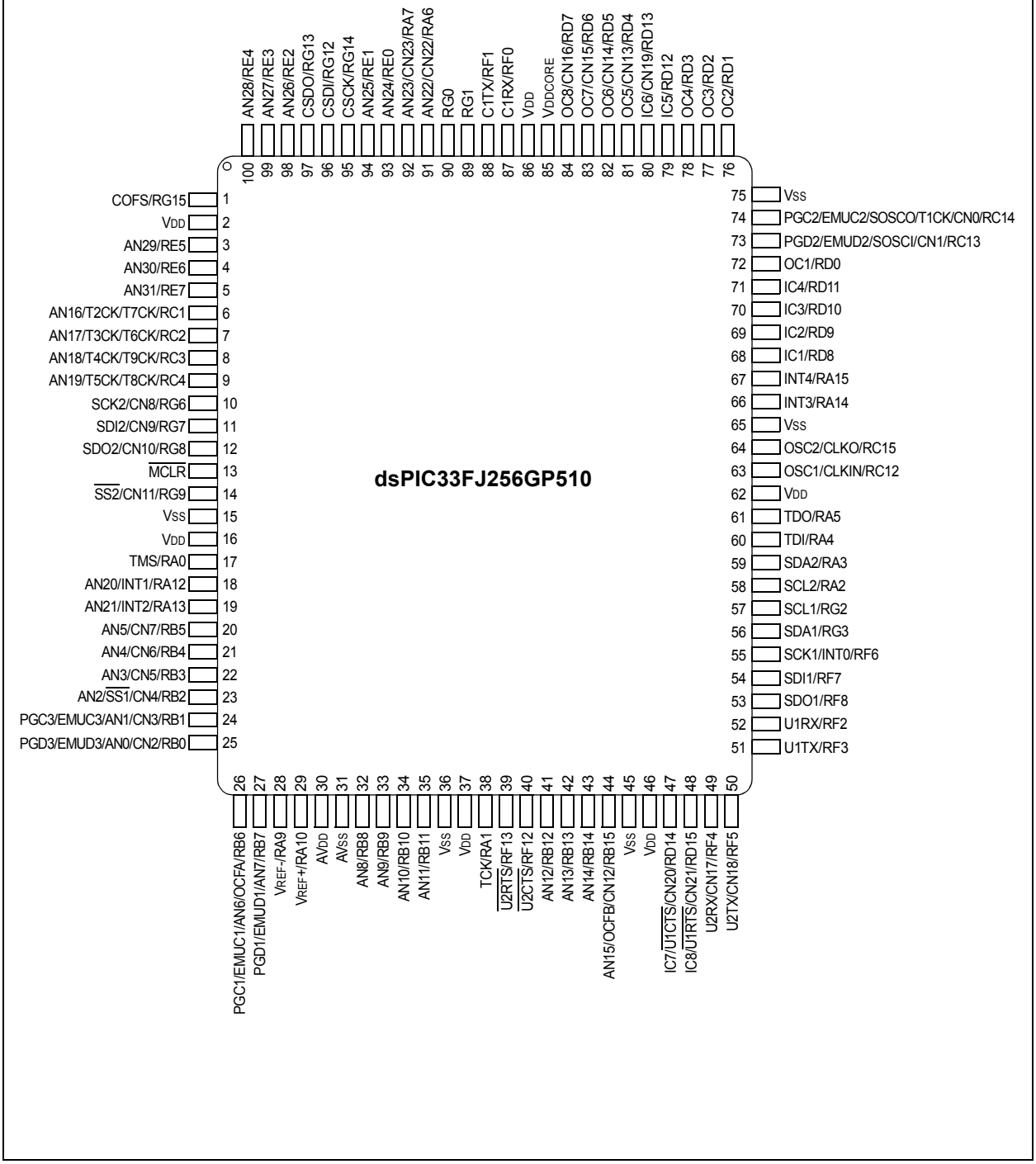
引脚图 （续）

100 引脚 TQFP



引脚图 （续）

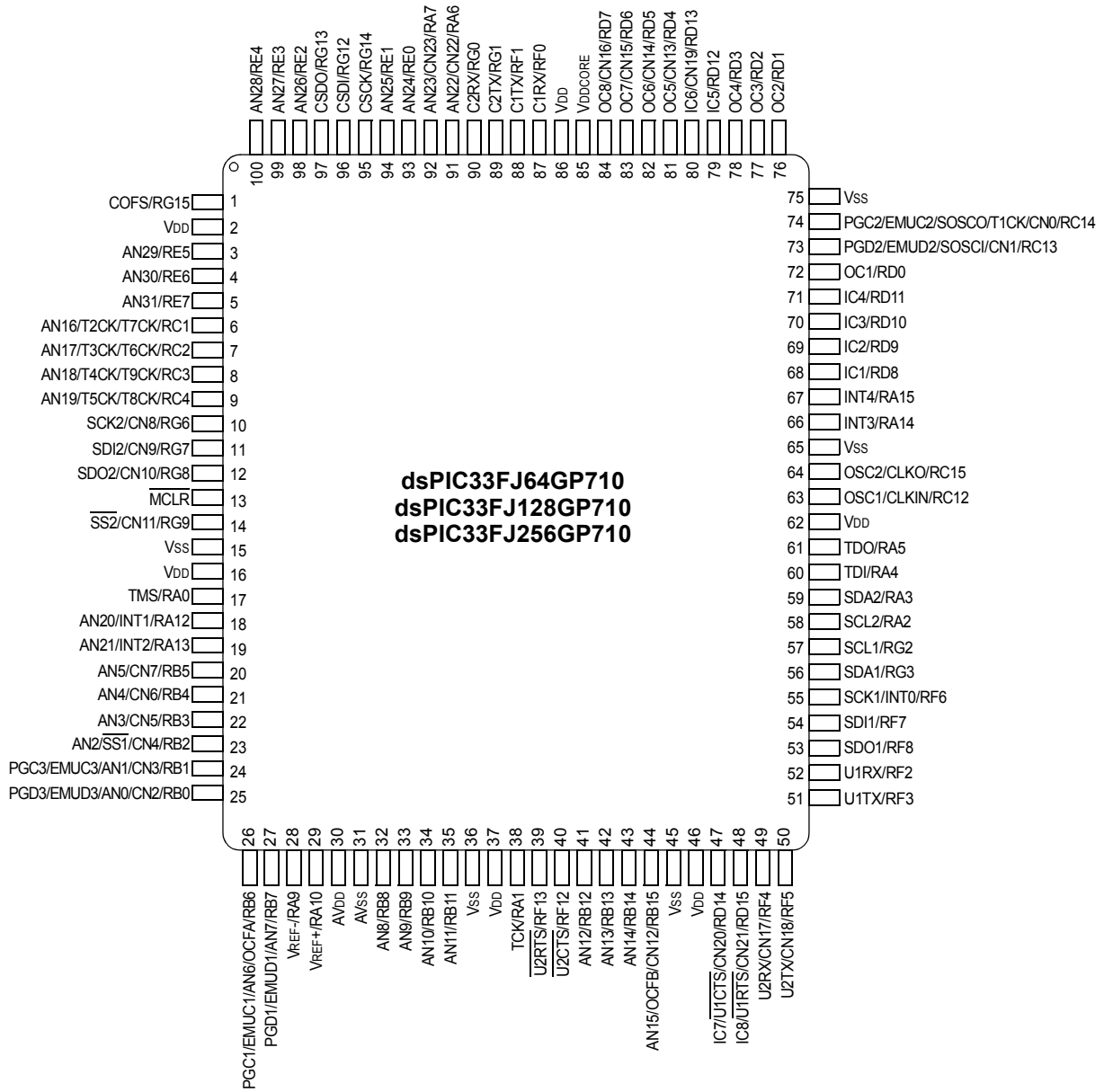
100 引脚 TQFP



© 2007 Microchip Technology Inc.

引脚图 （续）

100 引脚 TQFP



目录

dsPIC33F 产品系列 3
1.0 器件概述 13
2.0 CPU 17
3.0 存储器构成 29
4.0 闪存程序存储器 67
5.0 复位 73
6.0 中断控制器 79
7.0 直接存储器访问 （DMA） 125
8.0 振荡器配置 135
9.0 节能特性 143
10.0 I/O 端口 145
11.0 Timer1 147
12.0 Timer2/3、 Timer4/5、 Timer6/7 和 Timer8/9 149
13.0 输入捕捉 155
14.0 输出比较 157
15.0 串行外设接口 （SPI） 161
16.0 I^2C 169
17.0 通用异步收发器 （UART） 179
18.0 增强型 CAN （ECAN™）模块 187
19.0 数据转换器接口 （DCI）模块 217
20.0 10 位 /12 位模数转换器 （ADC） 231
21.0 特殊功能 245
22.0 指令集汇总 253
23.0 开发支持 261
24.0 电气特性 265
25.0 封装信息 305
附录 A： “PS”（原型样片）器件和最终产品器件之间的差异 311
附录 B： 版本历史 312
索引 313
Microchip 网站 319
变更通知客户服务 319
客户支持 319
读者反馈表 320
产品标识体系 321

致客户

我们旨在提供最佳文档供客户正确使用 Microchip 产品。为此，我们将不断改进出版物的内容和质量，使之更好地满足您的要求。出版物的质量将随新文档及更新版本的推出而得到提升。

如果您对本出版物有任何问题和建议，请通过电子邮件联系我公司 TRC 经理，电子邮件地址为 **CTRC@microchip.com**，或将本数据手册后附的 **《读者反馈表》**传真到 86-21-5407 5066。我们期待您的反馈。

最新数据手册

欲获得本数据手册的最新版本，请查询我公司的网站：

http://www.microchip.com

查看数据手册中任意一页下边角处的文献编号即可确定其版本。文献编号中数字串后的字母是版本号，例如：DS30000A是DS30000的 A 版本。

勘误表

现有器件可能带有一份勘误表，描述了实际运行与数据手册中记载内容之间存在的细微差异以及建议的变通方法。一旦我们了解到器件 / 文档存在某些差异时，就会发布勘误表。勘误表上将注明其所适用的硅片版本和文件版本。

欲了解某一器件是否存在勘误表，请通过以下方式之一查询：

- Microchip 网站 http://www.microchip.com
- 当地 Microchip 销售办事处 （见最后一页）

在联络销售办事处时，请说明您所使用的器件型号、硅片版本和数据手册版本 （包括文献编号）。

客户通知系统

欲及时获知 Microchip 产品的最新信息，请到我公司网站 **www.microchip.com** 上注册。

© 2007 Microchip Technology Inc.

1.0 器件概述

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

本文档涉及以下器件的具体信息：

- dsPIC33FJ64GP206
- dsPIC33FJ64GP306
- dsPIC33FJ64GP310
- dsPIC33FJ64GP706
- dsPIC33FJ64GP708
- dsPIC33FJ64GP710
- dsPIC33FJ128GP206
- dsPIC33FJ128GP306
- dsPIC33FJ128GP310
- dsPIC33FJ128GP706
- dsPIC33FJ128GP708
- dsPIC33FJ128GP710
- dsPIC33FJ256GP506
- dsPIC33FJ256GP510
- dsPIC33FJ256GP710

dsPIC33FJXXXGPX06/X08/X10 通用系列器件包括具有多种引脚数（64、80 和 100）、不同程序存储容量（64 KB、128 KB 和 256 KB）和不同 RAM 容量（8 KB、16 KB 和 30 KB）的器件。

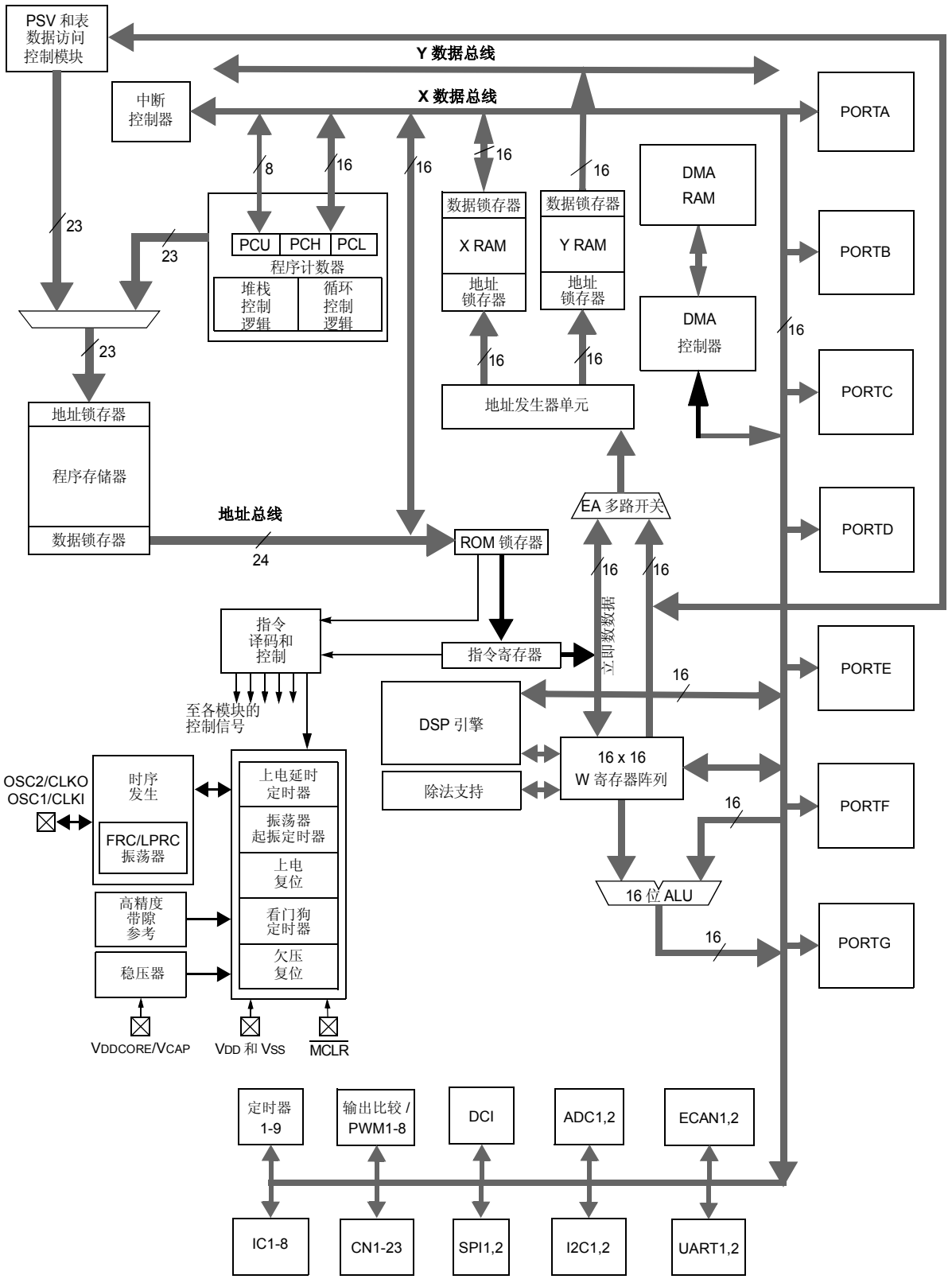
这使此系列器件适合于多种高性能数字信号控制应用。器件的引脚与 PIC24H 系列器件的引脚兼容，并且还与 dsPIC30F 系列器件高度兼容。这样便于根据应用对特定功能、计算资源和系统成本要求等方面的需求，在不同系列器件之间移植。

dsPIC33FJXXXGPX06/X08/X10 系列器件采用强大的 16 位架构，此架构将数字信号处理器（Digital Signal Processor，DSP）的计算能力与单片机（MCU）的控制特性无缝地集成在一起。这种集成的功能对于需要高速、重复计算和控制的应用非常理想。

DSP 引擎、两个 40 位累加器、支持除法运算的硬件、桶形移位寄存器、17 x 17 位乘法器、大的 16 位工作寄存器阵列和多种数据寻址模式，共同为 dsPIC33FJXXXGPX06/X08/X10 中央处理单元（Central Processing Unit，CPU）提供广泛的数学处理能力。灵活而确定的中断处理与功能强大的外设相结合，使得 dsPIC33FJXXXGPX06/X08/X10 器件非常适合控制应用。此外，直接存储器访问（DMA）允许数据在多个外设和专用 DMA RAM 之间进行无 CPU 开销的传输。可靠的现场可编程闪存程序存储器确保能对使用 dsPIC33FJXXXGPX06/X08/X10 器件的应用进行扩展。

图 1-1 给出了 dsPIC33FJXXXGPX06/X08/X10 系列器件中内核和外设模块的一般框图。表 1-1 列出了引脚图中显示的各引脚的功能。

图 1-1：　　dsPIC33FJXXXGPX06/X08/X10 一般框图



注：　　并非在所有器件的引脚配置中都实现了全部的引脚或功能。请参见引脚图了解每个器件上的特定引脚和功能的信息。

© 2007 Microchip Technology Inc.

表 1-1：　　引脚说明

引脚名称	引脚类型	缓冲器类型	说明
AN0-AN31	I	Analog	模拟输入通道。
AVDD	P	P	模拟模块的正电源。
AVSS	P	P	模拟模块的参考地。
CLKI CLKO	I O	ST/CMOS —	外部时钟源输入。总是与 OSC1 引脚功能相关联。 晶振输出。在晶振模式下，连接到晶体或谐振器。也可选择在 RC 和 EC 模式下用作 CLKO。总是与 OSC2 引脚功能相关联。
CN0-CN23	I	ST	输入电平变化通知输入。 可将所有输入软件编程为内部弱上拉。
COFS CSCK CSDI CSDO	I/O I/O I O	ST ST ST —	数据转换器接口帧同步引脚。 数据转换器接口串行时钟输入 / 输出引脚。 数据转换器接口串行数据输入引脚。 数据转换器接口串行数据输出引脚。
C1RX C1TX C2RX C2TX	I O I O	ST — ST —	ECAN1 总线接收引脚。 ECAN1 总线发送引脚。 ECAN2 总线接收引脚。 ECAN2 总线发送引脚。
PGD1/EMUD1 PGC1/EMUC1 PGD2/EMUD2 PGC2/EMUC2 PGD3/EMUD3 PGC3/EMUC3	I/O I I/O I I/O I	ST ST ST ST ST ST	编程 / 调试通信通道 1 使用的数据 I/O 引脚。 编程 / 调试通信通道 1 使用的时钟输入引脚。 编程 / 调试通信通道 2 使用的数据 I/O 引脚。 编程 / 调试通信通道 2 使用的时钟输入引脚。 编程 / 调试通信通道 3 使用的数据 I/O 引脚。 编程 / 调试通信通道 3 使用的时钟输入引脚。
IC1-IC8	I	ST	捕捉输入 1 至 8。
$\overline{\text{MCLR}}$	I/P	ST	主复位输入。此引脚为低电平有效的器件复位输入端。
OCFA OCFB OC1-OC8	I I O	ST ST —	比较故障 A 输入 （对于比较通道 1、 2、 3 和 4）。 比较故障 B 输入 （对于比较通道 5、 6、 7 和 8）。 比较输出 1 至 8。
OSC1 OSC2	I I/O	ST/CMOS —	晶振输入。配置为 RC 模式时为 ST 缓冲器输入；否则为 CMOS 输入。 晶振输出。在晶振模式下，连接到晶体或谐振器。也可选择在 RC 和 EC 模式下用作 CLKO。
RA0-RA7 RA9-RA10 RA12-RA15	I/O I/O I/O	ST ST ST	PORTA 是双向 I/O 端口。
RB0-RB15	I/O	ST	PORTB 是双向 I/O 端口。
RC1-RC4 RC12-RC15	I/O I/O	ST ST	PORTC 是双向 I/O 端口。
RD0-RD15	I/O	ST	PORTD 是双向 I/O 端口。
RE0-RE7	I/O	ST	PORTE 是双向 I/O 端口。
RF0-RF8 RF12-RF13	I/O	ST	PORTF 是双向 I/O 端口。
RG0-RG3 RG6-RG9 RG12-RG15	I/O I/O I/O	ST ST ST	PORTG 是双向 I/O 端口。
SCK1 SDI1 SDO1 $\overline{\text{SS1}}$ SCK2 SDI2 SDO2 $\overline{\text{SS2}}$	I/O I O I/O I/O I O I/O	ST ST — ST ST ST — ST	SPI1 的同步串行时钟输入 / 输出。 SPI1 数据输入。 SPI1 数据输出。 SPI1 从同步或帧脉冲 I/O。 SPI2 的同步串行时钟输入 / 输出。 SPI2 数据输入。 SPI2 数据输出。 SPI2 从同步或帧脉冲 I/O。

图注：　CMOS = CMOS 兼容输入或输出； Analog = 模拟输入
ST = CMOS 电平的施密特触发器输入； O = 输出； I = 输入； P = 电源

表 1-1：　引脚说明 （续）

引脚名称	引脚类型	缓冲器类型	说明
SCL1	I/O	ST	I2C1 的同步串行时钟输入 / 输出。
SDA1	I/O	ST	I2C1 的同步串行数据输入 / 输出。
SCL2	I/O	ST	I2C2 的同步串行时钟输入 / 输出。
SDA2	I/O	ST	I2C2 的同步串行数据输入 / 输出。
SOSCI	I	ST/CMOS	32.768 kHz 低功耗晶振输入；否则为 CMOS 输入。
SOSCO	O	—	32.768 kHz 低功耗晶振输出。
TMS	I	ST	JTAG 测试模式选择引脚。
TCK	I	ST	JTAG 测试时钟输入引脚。
TDI	I	ST	JTAG 测试数据输入引脚。
TDO	O	—	JTAG 测试数据输出引脚。
T1CK	I	ST	Timer1 外部时钟输入。
T2CK	I	ST	Timer2 外部时钟输入。
T3CK	I	ST	Timer3 外部时钟输入。
T4CK	I	ST	Timer4 外部时钟输入。
T5CK	I	ST	Timer5 外部时钟输入。
T6CK	I	ST	Timer6 外部时钟输入。
T7CK	I	ST	Timer7 外部时钟输入。
T8CK	I	ST	Timer8 外部时钟输入。
T9CK	I	ST	Timer9 外部时钟输入。
$\overline{\text{U1CTS}}$	I	ST	UART1 允许发送。
U1RTS	O	—	UART1 请求发送。
U1RX	I	ST	UART1 接收。
U1TX	O	—	UART1 发送。
$\overline{\text{U2CTS}}$	I	ST	UART2 允许发送。
U2RTS	O	—	UART2 请求发送。
U2RX	I	ST	UART2 接收。
U2TX	O	—	UART2 发送。
VDD	P	—	外设逻辑和 I/O 引脚的正电源。
VDDCORE	P	—	CPU 逻辑滤波电容连接。
VSS	P	—	逻辑和 I/O 引脚的参考地。
VREF+	I	Analog	模拟参考电压 （高电压）输入。
VREF-	I	Analog	模拟参考电压 （低电压）输入。

图注： CMOS = CMOS 兼容输入或输出； Analog = 模拟输入
ST = CMOS 电平的施密特触发器输入； O = 输出； I = 输入； P = 电源

© 2007 Microchip Technology Inc.

2.0 CPU

注：	本数据手册总结了dsPIC33FJXXXGPX06/X08/X10器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

dsPIC33FJXXXGPX06/X08/X10 CPU 模块采用 16 位（数据）的改进型哈佛架构，具有增强指令集，其中包括对 DSP 的强大支持。 CPU 具有 24 位指令字，指令字带有长度可变的操作码字段。程序计数器（Program Counter， PC）为 23 位宽，可以寻址最大 4M x 24 位的用户程序存储空间。实际实现的程序存储容量因器件而异。单周期指令预取机制可帮助维持吞吐量并使指令的执行具有预测性。除了改变程序流的指令、双字传送（MOV.D）指令和表指令以外，所有指令都在单个周期内执行。使用 DO 和 REPEAT 指令支持无开销的程序循环结构，这两条指令在任何时间都可以被中断。

dsPIC33FJXXXGPX06/X08/X10 器件在编程模型中有 16 个 16 位工作寄存器。每个工作寄存器都可以充当数据、地址或地址偏移量寄存器。第 16 个工作寄存器（W15）作为软件堆栈指针（Stack Pointer， SP），用于中断和调用。

dsPIC33FJXXXGPX06/X08/X10指令集具有两类指令：MCU 类指令和 DSP 类指令。这两类指令无缝地集成到单个 CPU 中。指令集包含多种寻址模式，指令的设计可使 C 编译器的效率达到最优。对于大多数指令，dsPIC33FJXXXGPX06/X08/X10 能够在每个指令周期内执行一次数据（或程序数据）存储器读取、一次工作寄存器（数据）读取、一次数据存储器写入以及一次程序（指令）存储器读取操作。因此，支持 3 操作数指令，允许在单个周期内执行 A + B = C 这样的操作。

图 2-1 给出了 CPU 的框图。图 2-2 给出了 dsPIC33FJXXXGPX06/X08/X10 的编程模型。

2.1 数据寻址概述

数据空间可以作为 32K 字或 64 KB 寻址，并被分成两块，称为 X 和 Y 数据存储区。每个存储块有各自独立的地址发生单元（Address Generation Unit， AGU）。MCU 类指令只通过 X 存储空间 AGU 进行操作，可将整个存储器映射作为一个线性数据空间访问。某些DSP指令通过X和Y的AGU进行操作以支持双操作数读操作，这样会将数据地址空间分成两个部分。X 和 Y 数据空间的边界视具体器件而定。

X 和 Y 地址空间都支持无开销循环缓冲区（模寻址模式）。模寻址省去了 DSP 算法的软件边界检查开销。此外，X AGU 的循环寻址可以用于任何 MCU 类指令。X AGU 还支持位反转寻址，大幅简化了基 2 FFT 算法对输入或输出数据的重新排序。

可以选择将数据存储空间映射的高 32 KB 映射到由 8 位程序空间可视性页（Program Space Visibility Page，PSVPAG）寄存器定义的任何 16K 程序字边界内的程序空间内。程序空间到数据空间的映射功能让任何指令都能象访问数据空间一样访问程序空间。数据空间还包括 2 KB 的 DMA RAM，它主要用于 DMA 数据传输，但也可用作通用 RAM。

2.2 DSP 引擎概述

DSP 引擎具有一个高速 17 位 x 17 位乘法器、一个 40 位 ALU、两个 40 位饱和累加器和一个 40 位双向桶形移位寄存器。该桶形移位寄存器能在单个周期内将一个 40 位的值右移或左移最多 16 位。DSP 指令可以无缝地与所有其他指令一起操作，且设计为能获得最佳实时性能。MAC 指令和其他相关指令可以在同一个周期内，同时完成从存储器中取两个数据操作数，将两个 W 寄存器相乘并累加，且可选择使结果饱和。这要求 RAM 数据存储空间对于这些指令拆分为两块，但对于所有其他指令保持线性。数据空间分块是通过将某些工作寄存器专用于每个地址空间，以透明和灵活的方式实现的。

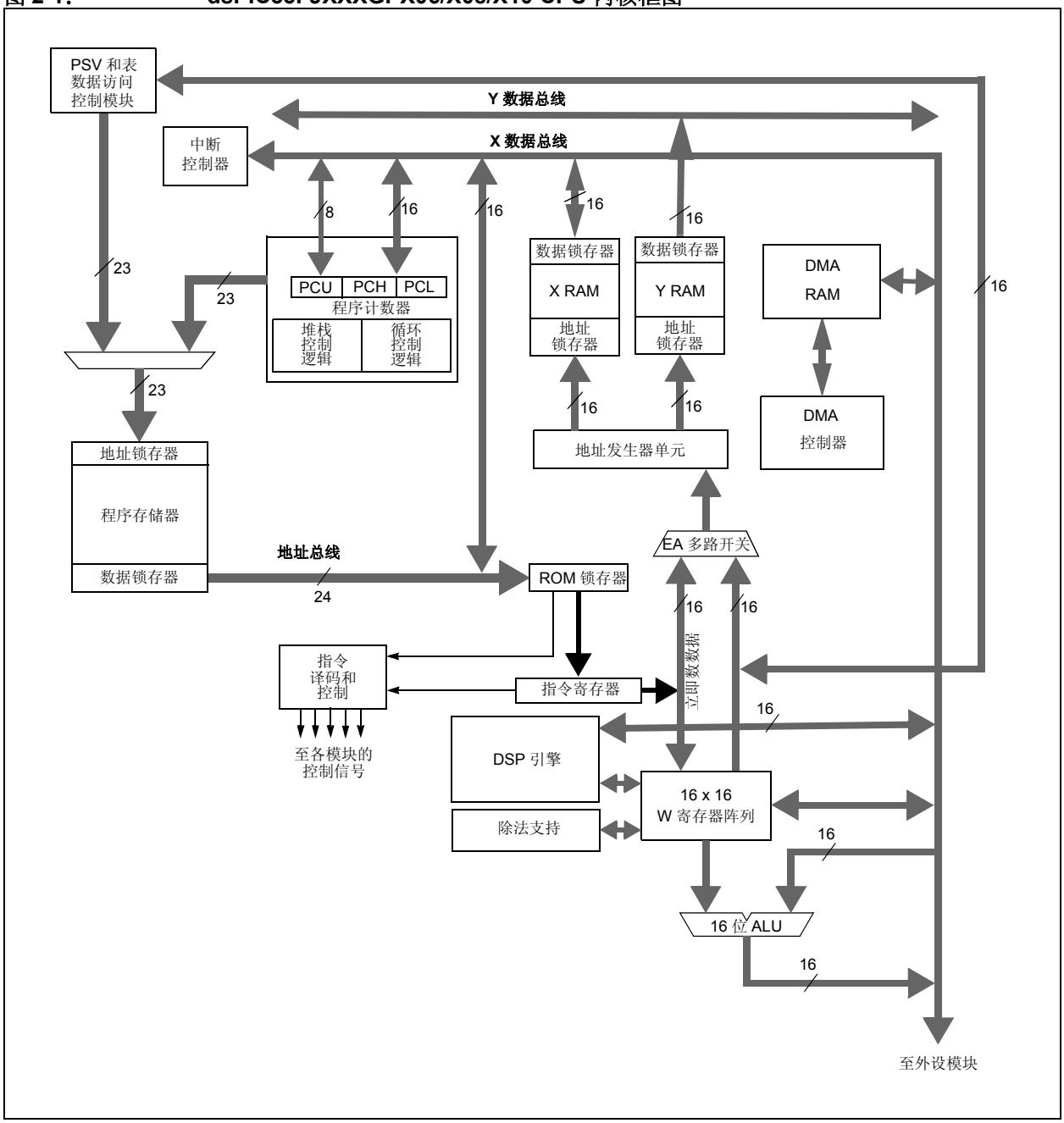
2.3 MCU 的特性

dsPIC33FJXXXGPX06/X08/X10 具有一个由 MCU ALU 和 DSP 引擎共用的 17 位 x 17 位单周期乘法器。此乘法器可以进行有符号、无符号和混合符号的乘法运算。使用 17 位 x 17 位乘法器进行 16 位 x 16 位乘法运算不仅允许您执行混合符号的乘法运算，而且对于 (-1.0) x (-1.0) 这样的特殊运算也可以得到准确结果。

dsPIC33FJXXXGPX06/X08/X10 支持小数和整数的 16/16 位和 32/16 位除法运算。所有的除法指令都是迭代操作。它们必须在一个 REPEAT 循环内执行，总执行时间为 19 个指令周期。在这 19 个周期的任一周期内可以中断除法运算而不会丢失数据。

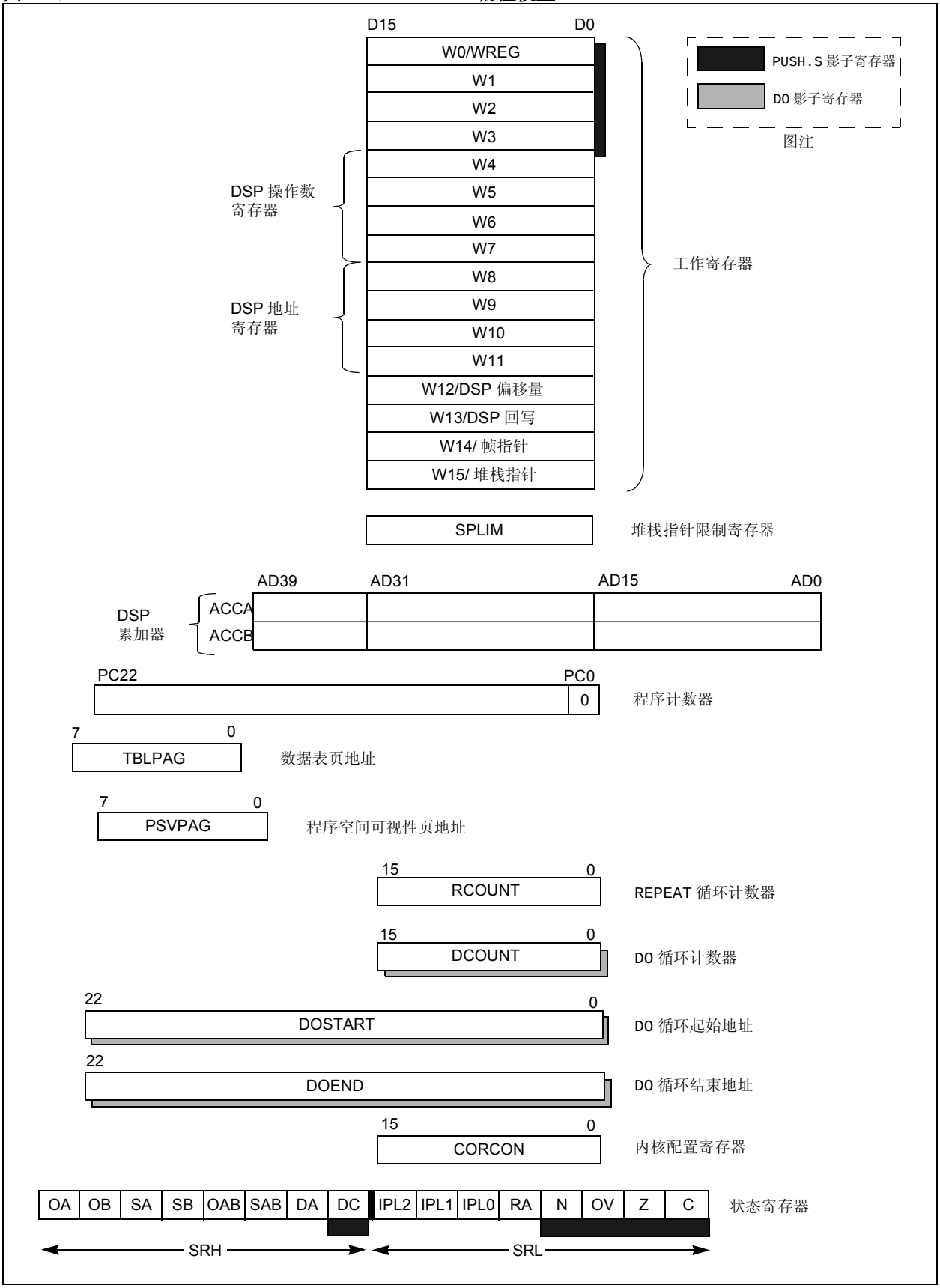
一个 40 位的桶形移位寄存器用于在单个周期内将数据左移或右移 16 位。 MCU 和 DSP 指令都可以使用该桶形移位寄存器。

图 2-1：　　dsPIC33FJXXXGPX06/X08/X10 CPU 内核框图



© 2007 Microchip Technology Inc.

图 2-2： dsPIC33FJXXXGPX06/X08/X10 编程模型



2.4 CPU 控制寄存器

CPU 控制寄存器包括：

- **SR：CPU 状态寄存器**
- **CORCON：内核控制寄存器**

寄存器 2-1： SR：CPU 状态寄存器

R-0	R-0	R/C-0	R/C-0	R-0	R/C-0	R -0	R/W-0
OA	OB	SA(1)	SB(1)	OAB	SAB	DA	DC
bit 15							bit 8

R/W-0(2)	R/W-0(3)	R/W-0(3)	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL<2:0>(2)			RA	N	OV	Z	C
bit 7							bit 0

图注：		
C = 只可清零位	R = 可读位	U = 未实现位，读为 0
S = 只可置 1 位	W = 可写位	-n = POR 时的值
1 = 置 1	0 = 清零	x = 未知

bit 15 **OA：** 累加器 A 溢出状态位
1 = 累加器 A 溢出
0 = 累加器 A 未溢出

bit 14 **OB：** 累加器 B 溢出状态位
1 = 累加器 B 溢出
0 = 累加器 B 未溢出

bit 13 **SA：** 累加器 A 饱和 “粘住”状态位 (1)
1 = 累加器 A 饱和或在某时已经饱和
0 = 累加器 A 未饱和

bit 12 **SB：** 累加器 B 饱和 “粘住”状态位 (1)
1 = 累加器 B 饱和或在某时已经饱和
0 = 累加器 B 未饱和

bit 11 **OAB：** OA 和 OB 组合的累加器溢出状态位
1 = 累加器 A 或 B 已经溢出
0 = 累加器 A 和 B 都未溢出

bit 10 **SAB：** SA 和 SB 组合的累加器 “粘住”状态位
1 = 累加器 A 或 B 饱和或在过去某时已经饱和
0 = 累加器 A 和 B 都未饱和

注： 此位可被读取或清零 （但不能置 1）。清零此位的同时将清零 SA 和 SB。

bit 9 **DA：** DO 循环活动位
1 = 正在进行 DO 循环
0 = 不在进行 DO 循环

注 1： 此位可被读取或清零 （但不能置 1）。

2： IPL<2:0> 位与 IPL<3> 位 （CORCON<3>）组合形成 CPU 中断优先级。如果 IPL<3> = 1，那么括号中的值表示 IPL。当 IPL<3> = 1 时，禁止用户中断。

3： 当 NSTDIS （INTCON1<15>） = 1 时， IPL<2:0> 状态位是只读的。

© 2007 Microchip Technology Inc.

寄存器 2-1：　　SR：CPU 状态寄存器 （续）

bit 8　　**DC**：MCU ALU 半进位 / 借位标志位
1 = 结果的第 4 个低位 （对于字节大小的数据）或第 8 个低位 （对于字大小的数据）发生了进位
0 = 结果的第 4 个低位 （对于字节大小的数据）或第 8 个低位 （对于字大小的数据）未发生进位

bit 7-5　　**IPL<2:0>**：CPU 中断优先级状态位 (2)
111 = CPU 中断优先级为 7 （15），禁止用户中断
110 = CPU 中断优先级为 6 （14）
101 = CPU 中断优先级为 5 （13）
100 = CPU 中断优先级为 4 （12）
011 = CPU 中断优先级为 3 （11）
010 = CPU 中断优先级为 2 （10）
001 = CPU 中断优先级为 1 （9）
000 = CPU 中断优先级为 0 （8）

bit 4　　**RA**：REPEAT 循环活动位
1 = 正在进行 REPEAT 循环
0 = 不在进行 REPEAT 循环

bit 3　　**N**：MCU ALU 负标志位
1 = 结果为负
0 = 结果为非负 （零或正值）

bit 2　　**OV**：MCU ALU 溢出标志位
此位用于有符号的算术运算（二进制补码）。它表示量值上的溢出，这种溢出将导致符号位改变状态。
1 = 有符号算术运算中发生溢出 （本次运算）
0 = 未发生溢出

bit 1　　**Z**：MCU ALU 全零标志位
1 = 影响 Z 位的任何运算在过去某时已将该位置 1
0 = 影响 Z 位的最近一次运算已将该位清零 （即运算结果非零）

bit 0　　**C**：MCU ALU 进位 / 借位标志位
1 = 结果的最高位发生了进位
0 = 结果的最高位未发生进位

注　1：此位可被读取或清零 （但不能置 1）。
2：IPL<2:0> 位与 IPL<3> 位 （CORCON<3>）组合形成 CPU 中断优先级。如果 IPL<3> = 1，那么括号中的值表示 IPL。当 IPL<3> = 1 时，禁止用户中断。
3：当 NSTDIS （INTCON1<15>） = 1 时， IPL<2:0> 状态位是只读的。

寄存器 2-2：　　CORCON：内核控制寄存器

U-0	U-0	U-0	R/W-0	R/W-0	R-0	R-0	R-0
—	—	—	US	EDT(1)	DL<2:0>		
bit 15							bit 8

R/W-0	R/W-0	R/W-1	R/W-0	R/C-0	R/W-0	R/W-0	R/W-0
SATA	SATB	SATDW	ACCSAT	IPL3(2)	PSV	RND	IF
bit 7							bit 0

图注：	C = 只可清零位		
R = 可读位	W = 可写位	-n = POR 时的值	1 = 置 1
0 = 清零	x = 未知	U = 未实现位，读为 0	

bit 15-13　**未实现：**读为 0

bit 12　**US：**DSP 乘法无符号 / 有符号控制位
1 = DSP 引擎执行无符号乘法运算
0 = DSP 引擎执行有符号乘法运算

bit 11　**EDT：**DO 循环提前终止控制位 (1)
1 = 在当前循环迭代结束时终止执行 DO 循环
0 = 无影响

bit 10-8　**DL<2:0>：**DO 循环嵌套层级状态位
111 = 正在进行 7 层 DO 循环嵌套
•
•
•
001 = 正在进行 1 层 DO 循环嵌套
000 = 正在进行 0 层 DO 循环嵌套

bit 7　**SATA：**AccA 饱和使能位
1 = 使能累加器 A 饱和
0 = 禁止累加器 A 饱和

bit 6　**SATB：**AccB 饱和使能位
1 = 使能累加器 B 饱和
0 = 禁止累加器 B 饱和

bit 5　**SATDW：**DSP 引擎的数据空间写饱和使能位
1 = 使能数据空间写饱和
0 = 禁止数据空间写饱和

bit 4　**ACCSAT：**累加器饱和模式选择位
1 = 9.31 饱和 （超饱和）
0 = 1.31 饱和 （正常饱和）

bit 3　**IPL3：**CPU 中断优先级状态位 3(2)
1 = CPU 中断优先级大于 7
0 = CPU 中断优先级等于或小于 7

bit 2　**PSV：**数据空间中程序空间可视性使能位
1 = 程序空间在数据空间中可视
0 = 程序空间在数据空间中不可视

bit 1　**RND：**舍入模式选择位
1 = 使能有偏 （常规）舍入
0 = 使能无偏 （收敛）舍入

bit 0　**IF：**整数或小数乘法器模式选择位
1 = 使能 DSP 乘法运算的整数模式
0 = 使能 DSP 乘法运算的小数模式

注　1：此位将总是读为 0。
2：IPL3 位与 IPL<2:0> 位 （SR<7:5>）组合形成 CPU 中断优先级。

© 2007 Microchip Technology Inc.

2.5 算术逻辑单元（ALU）

dsPIC33FJXXXGPX06/X08/X10 ALU 为 16 位宽，并能进行加法、减法、移位和逻辑运算。除非特别指明，算术运算一般采用二进制补码。根据不同的运算，ALU 可能会影响 SR 寄存器中的进位标志位（C）、全零标志位（Z）、负标志位（N）、溢出标志位（OV）和半进位标志位（DC）的值。在减法运算中， C 和 DC 状态位分别作为借位位和半借位位。

根据所使用的指令模式，ALU 可执行 8 位或 16 位运算。根据指令的寻址模式， ALU 运算的数据可以来自 W 寄存器阵列或数据存储器。同样，ALU 的输出数据可被写入 W 寄存器阵列或数据存储单元。

关于每条指令所影响的 SR 位的信息，请参见《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

dsPIC33FJXXXGPX06/X08/X10 CPU 融入了对乘法和除法的硬件支持。它带有专门的硬件乘法器以及支持 16 位除数除法的硬件。

2.5.1 乘法器

通过使用 DSP 引擎的高速 17 位 x 17 位乘法器，ALU 支持各种无符号、有符号或混合符号的 MCU 乘法运算：

1. 16 位 x 16 位有符号
2. 16 位 x 16 位无符号
3. 16 位有符号 x 5 位（立即数）无符号
4. 16 位无符号 x 16 位无符号
5. 16 位无符号 x 5 位（立即数）无符号
6. 16 位无符号 x 16 位有符号
7. 8 位无符号 x 8 位无符号

2.5.2 除法器

除法模块支持具有以下数据长度的 32 位 /16 位和 16 位 /16 位有符号和无符号整数除法运算：

1. 32 位有符号 /16 位有符号除法
2. 32 位无符号 /16 位无符号除法
3. 16 位有符号 /16 位有符号除法
4. 16 位无符号 /16 位无符号除法

所有除法指令的商都被放在 W0 中，余数放在 W1 中。16 位有符号和无符号 DIV 指令可为 16 位除数指定任一 W 寄存器（Wn），为 32 位被除数指定任意两个连续的 W 寄存器（W(m + 1):Wm）。除法运算中处理除数的每一位需要一个周期，因此 32 位 /16 位和 16 位 /16 位指令的执行周期数相同。

2.6 DSP 引擎

DSP 引擎由一个高速 17 位 x 17 位乘法器、一个桶形移位寄存器和一个 40 位加法器 / 减法器（两个目标累加器、舍入逻辑和饱和逻辑）组成。

dsPIC33FJXXXGPX06/X08/X10 采用单周期指令流架构；因此 DSP 引擎的工作不能与 MCU 指令流同时进行。但是，某些 MCU ALU 和 DSP 引擎资源可由同一条指令（如 ED 和 EDAC）同时使用。

DSP 引擎还可以执行固有的不需要其他数据的累加器 - 累加器操作。这些指令是 ADD、 SUB 和 NEG。

通过 CPU 内核控制寄存器（CORCON）中的各个位，可以对 DSP 引擎的操作进行多种选择，这些选择如下：

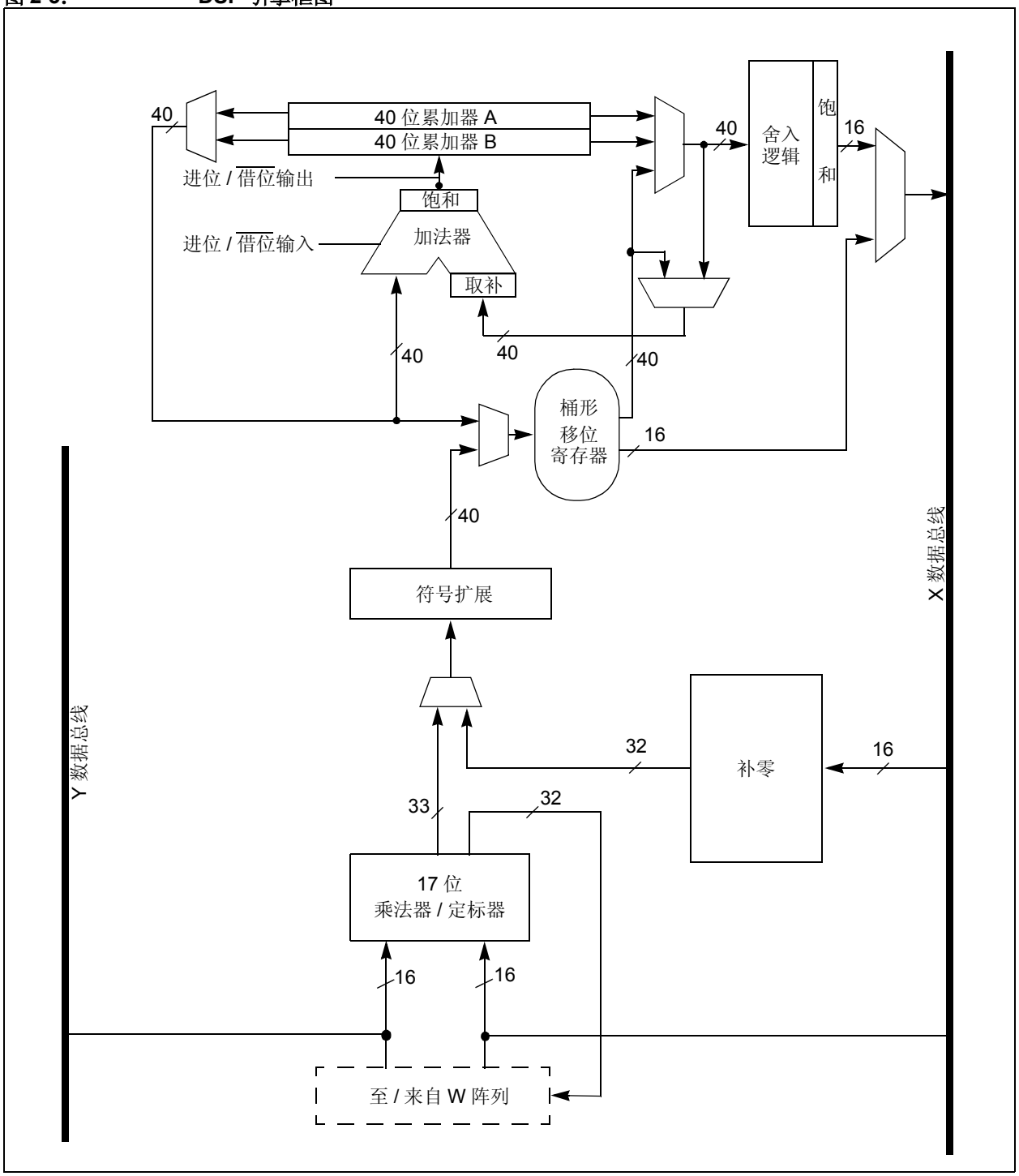
1. 小数或整数 DSP 乘法（IF）。
2. 有符号或无符号 DSP 乘法（US）。
3. 常规或收敛舍入（RND）。
4. AccA 自动饱和使能 / 禁止（SATA）。
5. AccB 自动饱和使能 / 禁止（SATB）。
6. 对于写数据存储器，自动饱和使能/禁止(SATDW)。
7. 累加器饱和模式选择（ACCSAT）。

表 2-1 给出了 DSP 指令的汇总。 DSP 引擎的框图如图 2-3 所示。

表 2-1： DSP 指令汇总

指令	代数运算	ACC 回写
CLR	A = 0	有
ED	$A = (x – y)^2$	无
EDAC	$A = A + (x – y)^2$	无
MAC	A = A + (x * y)	有
MAC	$A = A + x^2$	无
MOVSAC	A 中内容将不发生改变	有
MPY	A = x * y	无
MPY	$A = x^2$	无
MPY.N	A = – x * y	无
MSC	A = A – x * y	有

图 2-3：　　DSP 引擎框图



© 2007 Microchip Technology Inc.

2.6.1 乘法器

17 位 x 17 位乘法器可以进行有符号或无符号运算，其输出经过定标器进行换算后可支持 1.31 小数（Q31）或 32 位整数结果。无符号操作数经过零扩展后，送入乘法器输入值的第 17 位。有符号操作数经过符号扩展后，送入乘法器输入值的第 17 位。17 位 x 17 位乘法器 / 定标器的输出是 33 位值，它将被符号扩展为 40 位。整型数据的固有表示形式为有符号的二进制补码值，其中 MSb 定义为符号位。一般来说， N 位二进制补码整数的范围为 -2^{N-1} 到 $2^{N-1} – 1$。对于 16 位整数，数据范围为 -32768（0x8000）到 32767 （0x7FFF），包括 0 在内。对于 32 位整数，数据范围为 -2,147,483,648（0x8000 0000）到 2,147,483,647 （0x7FFF FFFF）。

当乘法器配置为小数乘法时，数据表示为二进制补码小数，其中 MSb 定义为符号位，小数点暗含在符号位之后 （QX 格式）。暗含小数点的 N 位二进制补码小数的范围为 -1.0 到（$1 – 2^{1-N}$）。对于 16 位小数，Q15 数据范围为 -1.0 （0x8000）到 0.999969482 （0x7FFF），包括 0 在内，其精度为 $3.01518x10^{-5}$。在小数模式下，16 x 16 乘法运算将产生 1.31 乘积，其精度为 $4.65661 x 10^{-10}$。

同一个乘法器还用来支持 MCU 乘法指令，包括整数的 16 位有符号、无符号和混合符号乘法。

MUL 指令可以使用字节或字长度的操作数。字节操作数将产生 16 位结果，而字操作数将产生 32 位结果，结果存放在 W 寄存器阵列的指定寄存器中。

2.6.2 数据累加器和加法器 / 减法器

数据累加器包含一个 40 位加法器 / 减法器，它带有自动符号扩展逻辑。它可以选择两个累加器 （A 或 B）之一作为其累加前的源累加器和累加后的目标累加器。对于 ADD 和 LAC 指令，可选择通过桶形移位器在累加之前将要累加或装入的数据进行换算。

2.6.2.1 加法器 / 减法器、溢出和饱和

加法器 / 减法器是一个 40 位加法器，一侧输入可以选择为零，而另一侧输入可以是原数据或求补后的数据。对于加法，进位 / 借位输入为高电平有效，其他输入是原数据 （没有求补的）；对于减法，进位 / 借位输入为低电平有效，其他输入是求补后的数据。加法器 / 减法器产生溢出状态位 SA/SB 和 OA/OB，这些状态位被锁存在状态寄存器中并在其中得到反映：

- 从 bit 39 溢出：这是灾难性溢出，会破坏累加器的符号位。
- 溢出到警戒位（bit 32 到 bit 39）：这是可恢复的溢出。每当警戒位彼此不完全一致时，就将把这个状态位置 1。

加法器有一个额外的饱和模块，如果选取的话，饱和模块将控制累加器的数据饱和。饱和模块使用加法器的结果、上述的溢出状态位、SAT<A:B>（CORCON<7:6>）和 ACCSAT （CORCON<4>）模式控制位，来确定何时饱和、达到何值为饱和。

状态寄存器中有 6 个支持饱和和溢出的位，它们是：

1. OA：
   AccA 溢出到警戒位
2. OB：
   AccB 溢出到警戒位
3. SA：
   AccA 已饱和 （bit 31 溢出并饱和）
   或
   AccA 溢出到警戒位并饱和（bit 39 溢出并饱和）
4. SB：
   AccB 已饱和 （bit 31 溢出并饱和）
   或
   AccB 溢出到警戒位并饱和（bit 39 溢出并饱和）
5. OAB：
   OA 和 OB 的逻辑或 （OR）
6. SAB：
   SA 和 SB 的逻辑或 （OR）

每次数据通过加法器 / 减法器，就会修改 OA 和 OB 位。置 1 时，它们表明最近的操作已溢出到累加器警戒位（bit 32 到 bit 39）。如果 OA 和 OB 位置 1 而且 INTCON1 寄存器中相应的溢出陷阱标志允许位 （OVATE 和 OVBTE）置 1 的话，还可以选择用 OA 和 OB 位产生算术警告陷阱 （见**第 6.0 节 “中断控制器”**）。这使得用户能够立即采取措施，例如，校正系统增益。

每次数据通过加法器 / 减法器，就会修改 SA 和 SB 位，但用户只能对它们进行清零。置 1 时，它们表明累加器已溢出其最大范围（对于 32 位饱和是 bit 31，而 40 位饱和是 bit 39），将发生饱和（如果饱和使能的话）。如果没有使能饱和，SA 和 SB 置 1 默认为 bit 39 溢出，以此指示产生了灾难性溢出。如果 INTCON1 寄存器中的 COVTE 位置 1，当饱和被禁止时， SA 和 SB 位将产生算术警告陷阱。

在状态寄存器 （SR）中，对于溢出和饱和状态位，可以将 OA 和 OB 的逻辑或形成 OAB 位，将 SA 和 SB 的逻辑或形成 SAB 位。这样，只需检查状态寄存器中的一个位，程序员就能判断是否有累加器溢出；检查状态寄存器中的另一个位，就可以判断是否有累加器饱和。对于通常要使用两个累加器的复数运算而言，这很有用。

器件支持三种饱和和溢出模式：

1. bit 39 溢出和饱和：
   当发生 bit 39 溢出和饱和时，饱和逻辑将最大的正 9.31 值 （0x7FFFFFFFFF）或最小的负 9.31 值（0x8000000000）装入目标累加器。SA 或 SB 位置 1 并保持直到被用户清零。这称为 “超饱和”，为错误数据或不可预期的算法问题（例如，增益计算）提供了保护机制。
2. bit 31 溢出和饱和：
   当发生 bit 31 溢出和饱和时，饱和逻辑将最大的正 1.31 值 （0x007FFFFFFF）或最小的负 1.31 值 （0x0080000000）装入目标累加器。 SA 或 SB 位置 1 并保持直到被用户清零。当这种饱和模式生效时，不使用警戒位 （因此 OA、 OB 或 OAB 位不会被置 1）。
3. bit 39 灾难性溢出：
   加法器的 bit 39 溢出状态位用来将 SA 或 SB 位置 1 ；这两位置 1 后，将保持状态直到被用户清零。不进行饱和操作，允许累加器溢出（破坏其符号位）。如果 INTCON1 寄存器中的 COVTE 位置 1，灾难性溢出会导致一个陷阱异常。

2.6.2.2 累加器 “回写”

MAC 类指令（MPY、MPY.N、ED 和 EDAC 除外）可以选择将累加器高位字 （bit 16 到 bit 31）的舍入形式写入数据存储空间，前提是当前指令不对该累加器进行操作。通过 X 总线寻址组合的 X 和 Y 地址空间，执行回写操作。支持以下寻址模式：

1. W13，寄存器直接寻址：
   非操作目标的累加器的舍入内容以1.15小数形式写入 W13。
2. [W13]+=2，执行后递增的寄存器间接寻址：
   非操作目标的累加器的舍入内容以1.15小数形式写入 W13 指向的地址。然后 W13 递增 2（对于字写入）。

2.6.2.3 舍入逻辑

舍入逻辑是一个组合模块，在累加器写（存储）过程中执行常规的（有偏）或收敛的（无偏）舍入功能。舍入模式由 CORCON 寄存器中 RND 位的状态决定。它会产生一个 16 位的 1.15 数据值，该值被送到数据空间写饱和逻辑。如果指令不指明舍入，就会存储一个截取的 1.15 数据值，简单地丢弃低位字。

常规舍入取累加器的 bit 15，对它进行零扩展并将扩展后的值加到 ACCxH 字（累加器的 bit 16 到 bit 31）。如果 ACCxL 字 （累加器的 bit 0 到 bit 15）在 0x8000 和 0xFFFF 之间 （包括 0x8000），则 ACCxH 递增 1。如果 ACCxL 在 0x0000 和 0x7FFF 之间，则 ACCxH 不变。此算法的结果经过一系列随机舍入操作，值会稍稍偏大 （正偏）。

除非 ACCxL 等于 0x8000，否则收敛的（或无偏）舍入操作方式与常规舍入相同。在这种情况下，要对 ACCxH 的最低位 （累加器的 bit 16）进行检测。如果它为 1，ACCxH 递增 1。如果它为 0，ACCxH 不变。假设 bit 16 本身是随机的，这样的机制将消除任何可能累加的舍入偏差。

通过 X 总线， SAC 和 SAC.R 指令将目标累加器内容的截取 （SAC）或舍入 （SAC.R）形式存入数据存储空间（这受数据饱和的影响，请参见**第 2.6.2.4 节 “数据空间写饱和”**）。对于 MAC 类指令，累加器回写操作将以同样的方式进行，通过 X 总线寻址组合的 MCU （X 和 Y）数据空间。对于此类指令，数据始终要进行舍入。

© 2007 Microchip Technology Inc.

2.6.2.4 数据空间写饱和

除了加法器 / 减法器饱和，对数据空间进行写操作也会饱和，但不会影响源累加器的内容。数据空间写饱和逻辑块接受来自舍入逻辑块的一个 16 位的 1.15 小数值作为输入，还接受来自源 （累加器）和 16 位舍入加法器的溢出状态。这些输入经过组合用来选择适当的 1.15 小数值作为输出，写入数据存储空间中。

如果 CORCON 寄存器中的 SATDW 位置 1，将检测（经过舍入或截取后的）数据是否溢出，并进行相应的调整。如果输入数据大于 0x007FFF，则写入数据存储器中的数据被强制为最大的正 1.15 值， 0x7FFF。如果输入数据小于0xFF8000，则写入存储器中的数据被强制为最小的负 1.15 值，0x8000。源累加器的最高位（bit 39）用来决定被检测的操作数的符号。

如果 CORCON 寄存器中的 SATDW 位没有置 1，则输入数据都将通过，在任何情况下都不会被修改。

2.6.3 桶形移位寄存器

桶形移位寄存器在单个周期内可将数据算术或逻辑右移或左移最多 16 位。源操作数可以是两个 DSP 累加器中的任何一个或 X 总线（支持寄存器或存储器中数据的多位移位）。

移位寄存器需要一个有符号二进制值，用来确定移位操作的幅度（位数）和方向。正值将操作数右移。负值则将操作数左移。值为 0 则不改变操作数。

桶形移位寄存器为 40 位宽，于是，它为 DSP 移位操作提供了 40 位的结果，而为 MCU 移位操作提供 16 位的结果。来自 X 总线的数据在桶形移位寄存器中的存放方式是：右移则数据存放在 bit 16 到 bit 31，左移则存放在 bit 0 到 bit 15。

注：

© 2007 Microchip Technology Inc.

3.0 存储器构成

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

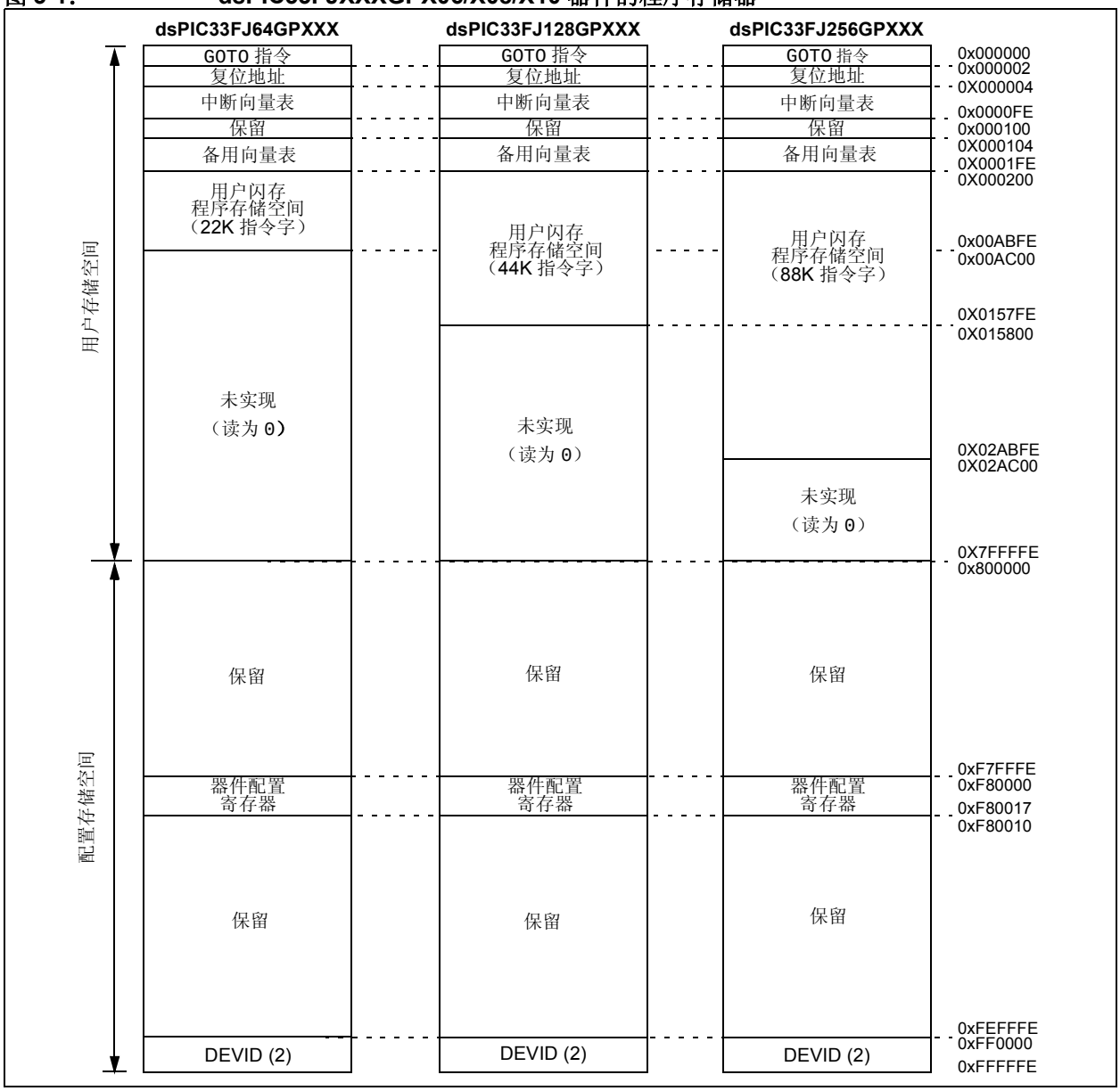
dsPIC33FJXXXGPX06/X08/X10 架构具有独立的程序和数据存储空间以及总线。这一架构同时还允许在代码执行过程中从数据空间直接访问程序存储器。

3.1 程序地址空间

dsPIC33FJXXXGPX06/X08/X10 器件的程序地址存储空间可存储 4M 个指令字。可通过由程序执行过程中 23 位程序计数器（PC）和**第 3.6 节 “程序存储空间与数据存储空间的接口”**中所述的表操作或数据空间重映射得到的 24 位值寻址这一空间。

用户只能访问程序存储空间的低半地址部分（地址范围为 0x000000 至 0x7FFFFF）。使用 TBLRD/TBLWT 指令时，情况有所不同，这两条指令采用 TBLPAG<7> 以允许访问配置存储空间中的配置位和器件 ID。图 3-1 给出了 dsPIC33FJXXXGPX06/X08/X10 器件的存储器使用情况。

图 3-1：　　dsPIC33FJXXXGPX06/X08/X10 器件的程序存储器



© 2007 Microchip Technology Inc.

3.1.1 程序存储器构成

程序存储空间由可字寻址的块构成。虽然它被视为24位宽，但将程序存储器的每个地址视作一个低位字和一个高位字的组合更加合理，其中高位字的高字节部分没有实现。低位字的地址始终为偶数，而高位字的地址为奇数（图 3-2）。

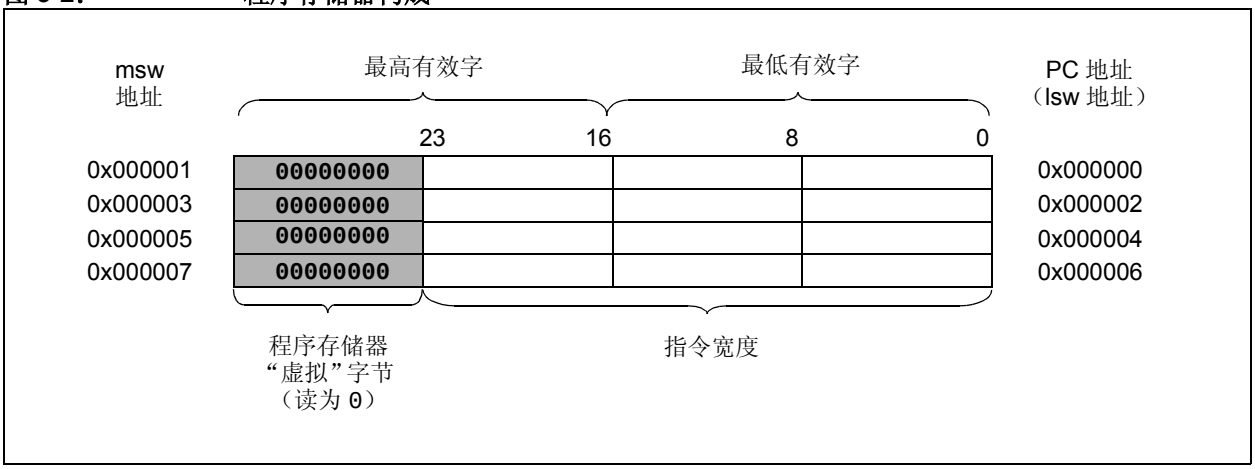
程序存储器地址始终在低位字处按字对齐，并且在代码执行过程中地址将递增或递减 2。这种寻址模式与数据存储空间寻址兼容，且为访问程序存储空间中的数据提供了可能。

3.1.2 中断和陷阱向量

所有 dsPIC33FJXXXGPX06/X08/X10 器件中从 0x00000 到 0x000200 之间的地址空间都是保留的，用来存储硬编码的程序执行向量。提供了一个硬件复位向量将代码执行从器件复位时 PC 的默认值重新定位到代码实际起始处。用户可在地址 0x000000 处编写一条 GOTO 指令以将代码的实际起始地址设置为 0x000002。

dsPIC33FJXXXGPX06/X08/X10 器件也具有两个中断向量表，地址分别为从 0x000004 到 0x0000FF 和 0x000100 到 0x0001FF。这两个向量表允许使用不同的中断服务程序（Interrupt Service Routines，ISR）处理每个器件中断源。关于中断向量表更详细的讨论，请参见**第 6.1 节 “中断向量表”**。

图 3-2： 程序存储器构成



3.2 数据地址空间

dsPIC33FJXXXGPX06/X08/X10 CPU 具有独立的 16 位宽数据存储空间。使用独立的地址发生单元（AGU）对数据空间执行读写操作。图 3-3 到图 3-5 给出了带有不同 RAM 大小的器件的数据存储器映射情况。

数据存储空间中的所有有效地址 （Effective Addresse，EA）均为 16 位宽，并且指向数据空间内的字节。这种构成方式使得数据空间的地址范围为 64 KB 或 32K 字。数据存储空间的低半地址部分 （即当 EA<15> = 0 时）用作实现的存储单元，而高半地址部分 （EA<15> = 1）则保留为程序空间可视性 （Program Space Visibility，PSV）区域 （见**第 3.6.3 节 “使用程序空间可视性读程序存储器中的数据”**）。

dsPIC33FJXXXGPX06/X08/X10 器件共实现了最大 30 KB 的数据存储空间。如果 EA 指向了该区域以外的存储单元，则将返回一个全零的字或字节。

3.2.1 数据空间宽度

数据存储空间组织为可字节寻址的 16 位宽的块。在数据存储器和寄存器中的数据是以16位字为单位对齐的，但所有数据空间 EA 都将解析为字节。每个字的低字节（Least Significant Byte，LSB）部分具有偶地址，而高字节 （Most Significant Byte， MSB）部分则具有奇地址。

3.2.2 数据存储器构成和对齐方式

为维持与 PIC® MCU 器件的后向兼容性和提高数据存储空间的使用效率， dsPIC33FJXXXGPX06/X08/X10 指令集同时支持字和字节操作。字节访问会在内部对按字对齐的存储空间的所有有效地址进行计算调整。例如，对于执行后修改寄存器间接寻址模式 [Ws++] 的结果，字节操作时，内核将其识别为值 Ws + 1，而字操作时，内核将其识别为值 Ws + 2。

使用任何 EA 的 LSb 来确定要选取的字节，数据字节读取将读取包含字节的整个字。选定的字节被放在数据总线的 LSb 处。这就是说，数据存储器和寄存器被组织为两个并行的字节宽的实体，它们共享 （字）地址译码，但写入线独立。数据字节写操作只写阵列或寄存器中与字节地址匹配的那一侧。

所有字访问必须按偶地址对齐。不支持不对齐的字数据取操作，所以在混合字节和字操作时，或者从 8 位 MCU 代码移植时，必须要小心。如果试图进行不对齐的读或写操作，将产生地址错误陷阱。如果在读操作时产生错误，正在执行的指令将完成；而如果在写操作时产生错误，指令仍将执行，但不会进行写入。无论是哪种情况，都会产生陷阱，从而系统和 / 或用户能够检查地址错误发生之前的机器状态。

所有装入 W 寄存器的字节都将装入 W 寄存器的低字节（LSB）， W 寄存器的高字节 （MSB）不变。

提供了一条符号扩展 （SE）指令，允许用户把 8 位有符号数据转换为 16 位有符号值。或者，对于 16 位无符号数据，用户可以通过在适当地址处执行一条零扩展（ZE）指令清零任何 W 寄存器的 MSb。

3.2.3 SFR 空间

Near数据空间的前2 KB存储单元（从0x0000到0x07FF）主要被特殊功能寄存器 （Special Function Registers，SFR）占用。 dsPIC33FJXXXGPX06/X08/X10 内核和外设模块使用这些寄存器来控制器件的工作。

SFR 分布在受其控制的模块中，通常一个模块会使用一组 SFR。大部分 SFR 空间包含未用的存储单元，它们读为 0。表 3-1 到表 3-32 给出了所有实现的 SFR 及其地址的完整列表。

注：	不同器件的实际外设功能集和中断也各不相同。关于特定器件的信息，请参见相应器件的数据表和引脚图。

3.2.4 NEAR 数据空间

在 0x0000 和 0x1FFF 之间的 8 KB 的区域被称为 Near 数据空间。可以使用所有存储器直接寻址指令中的13位绝对地址字段直接寻址这一空间中的存储单元。此外，还可以使用 MOV 指令寻址整个数据空间，支持使用 16 位地址字段的存储器直接寻址模式或使用工作寄存器作为地址指针的间接寻址模式。

© 2007 Microchip Technology Inc.

图 3-3：　　带有 8 KB RAM 的 dsPIC33FJXXXGPX06/X08/X10 器件的数据存储器映射

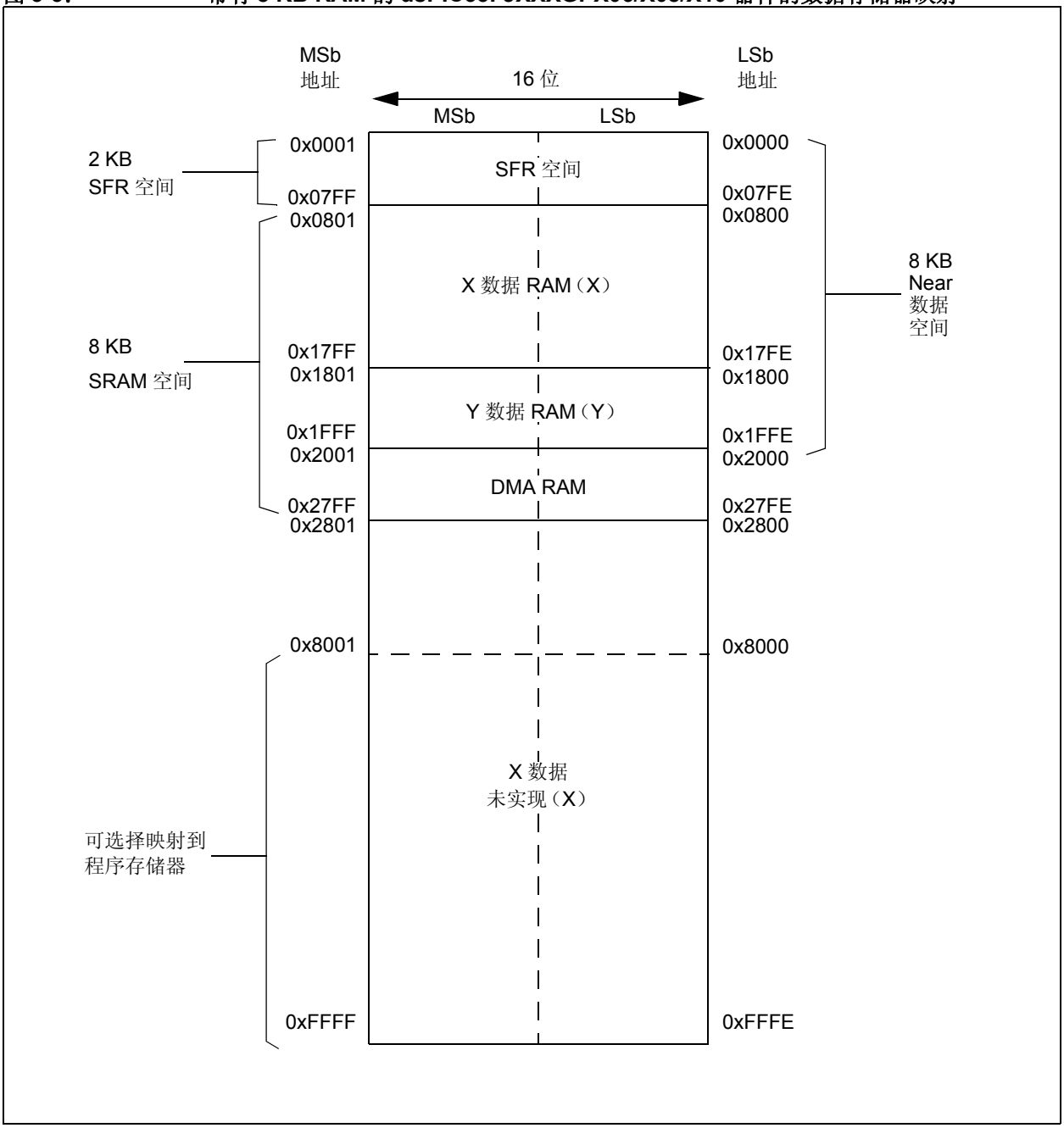
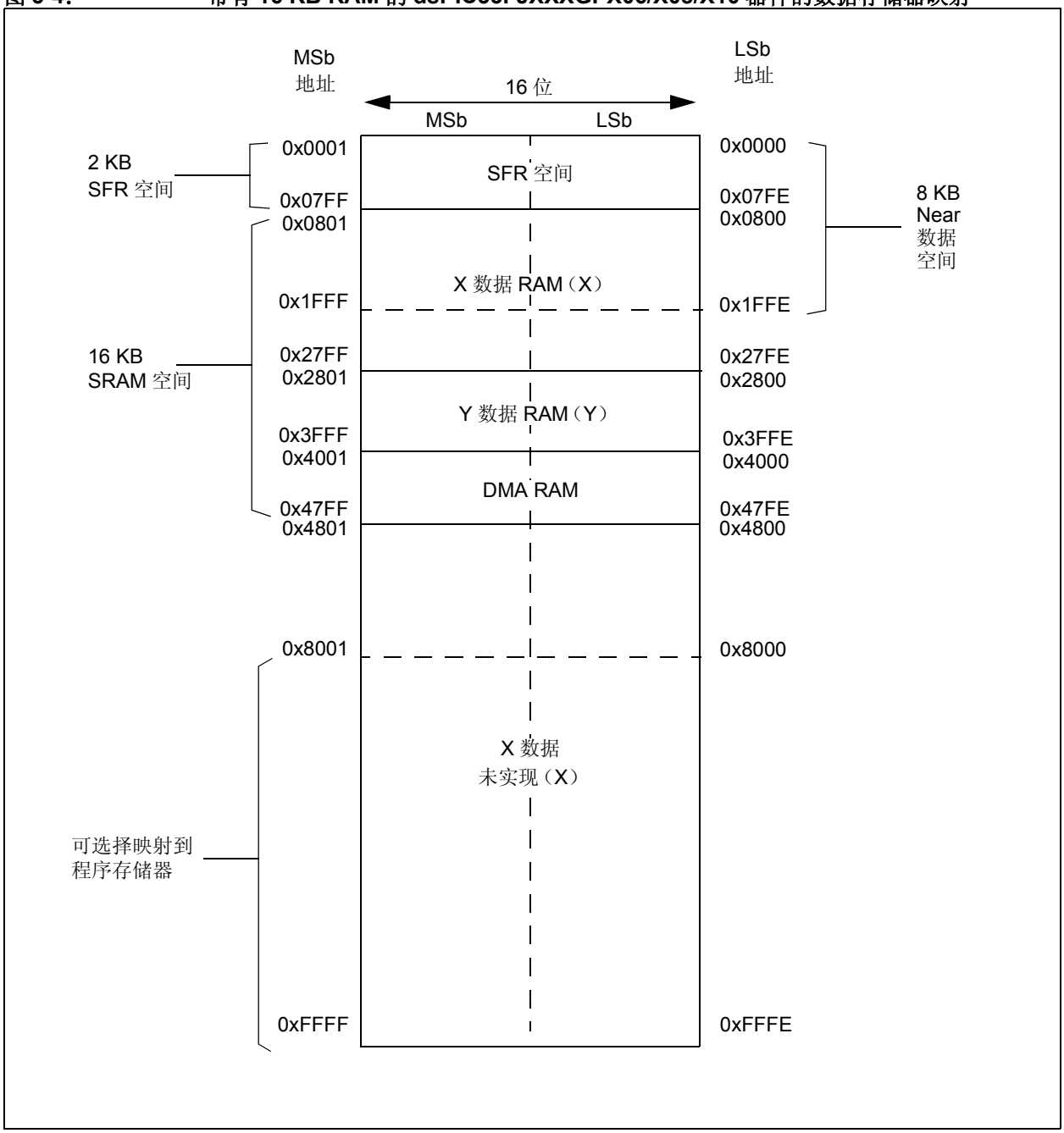
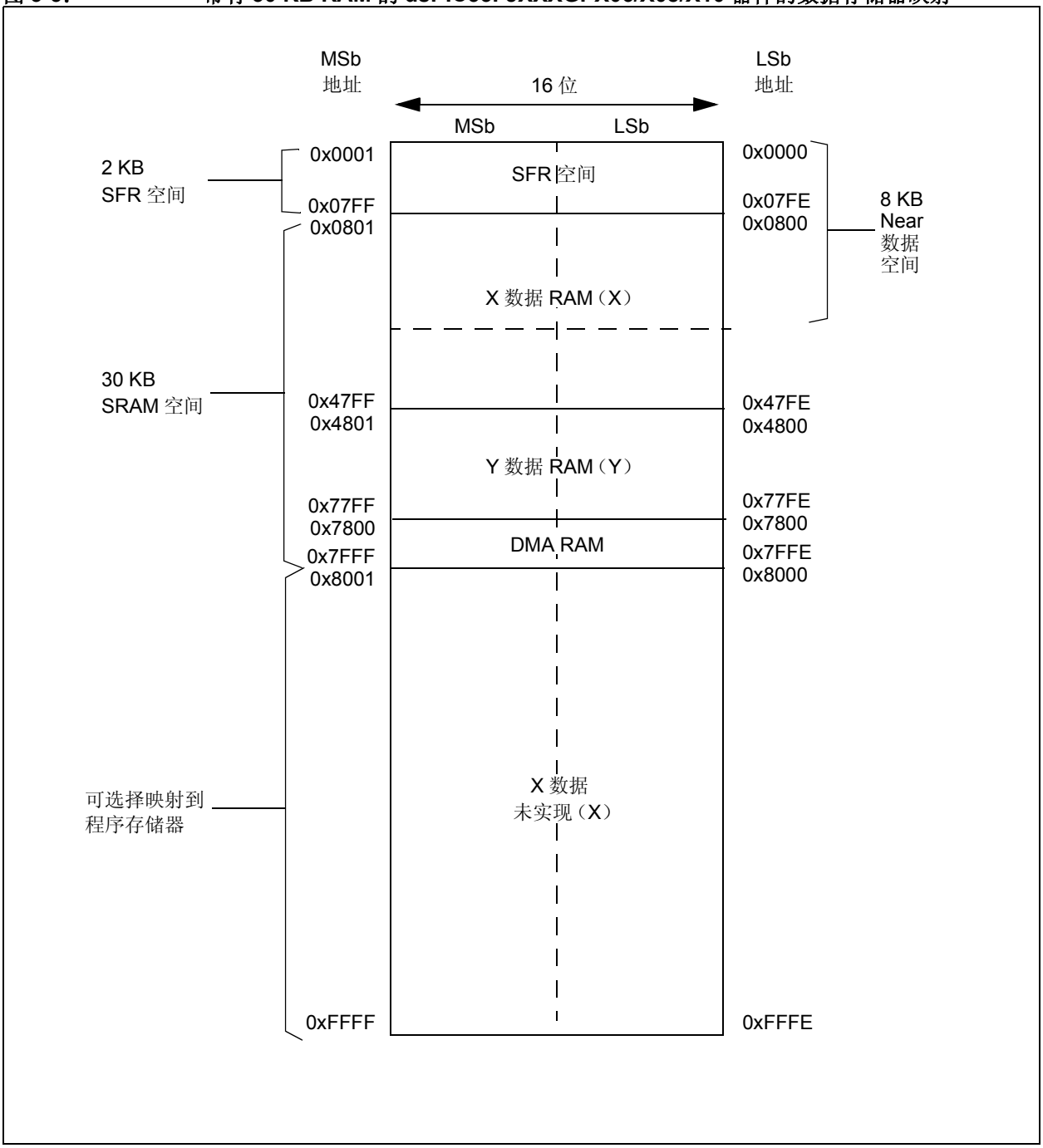


图 3-4：　　带有 16 KB RAM 的 dsPIC33FJXXXGPX06/X08/X10 器件的数据存储器映射



© 2007 Microchip Technology Inc.

图 3-5：　　带有 30 KB RAM 的 dsPIC33FJXXXGPX06/X08/X10 器件的数据存储器映射



3.2.5 X 和 Y 数据空间

内核有两个数据空间 X 和 Y。这些数据空间可以看作是独立的 （对于一些 DSP 指令），或者看作是统一的线性地址范围 （对于 MCU 指令）。使用两个地址发生单元 （AGU）和独立的数据总线来访问这两个数据空间。此特性允许某些指令同时从 RAM 中取两个字，因此提高了某些DSP算法的执行效率，如有限冲激响应（Finite Impulse Response， FIR）滤波算法和快速傅立叶变换（Fast Fourier Transform， FFT）。

X 数据空间可用于所有指令，并且支持所有寻址模式。X 数据空间的读数据总线和写数据总线是独立的。所有将数据空间视为组合的X和Y地址空间的指令均将X读数据总线作为读数据路径。X 读数据总线也可为双操作数 DSP 指令 （MAC 类）的 X 数据预取路径。

MAC 类指令（CLR、ED、EDAC、MAC、MOVSAC、MPY、MPY.N和MSC）将同时使用X数据空间和Y数据空间，从而提供两条可同时对数据进行读操作的路径。

X 和 Y 数据空间都支持所有指令的模寻址，但要受到寻址模式的限制。位反转寻址模式只是在写 X 数据空间时才支持。

所有数据存储器写操作 （包括 DSP 指令中的数据存储器写操作）均把数据空间视为组合的 X 和 Y 地址空间。X 和 Y 数据空间的分界取决于具体的器件，且不能由用户编程。

所有有效地址均为16位宽并且指向数据空间内的字节。因此，数据空间地址范围为 64 KB 或 32K 字，尽管不同器件上实际实现的存储单元有所不同。

3.2.6 DMA RAM

每个 dsPIC33FJXXXGPX06/X08/X10 器件包含 2 KB 的双端口 DMA RAM，位于 Y 数据空间的末端。DMA RAM 空间是Y数据RAM的一部分，其中的存储单元可被CPU和 DMA 控制器模块同时访问。 DMA 控制器使用 DMA RAM 存储使用 DMA 传输到各个外设的数据，以及使用 DMA 从各个外设传输进来的数据。 DMA 控制器可以在不占用 CPU 周期的情况下访问 DMA RAM。

当CPU和DMA控制器尝试同时写同一个DMA RAM单元时，硬件确保 CPU 具有优先访问权。因此， DMA RAM 提供了传输 DMA 数据的可靠方式，而无须暂停 CPU。

注：	在不需要使用 DMA 功能的应用中， DMA RAM 可作为通用数据存储区使用。

© 2007 Microchip Technology Inc.

表 3-1：　CPU 内核寄存器映射

SFR 名称	SFR 地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
WREG0	0000	工作寄存器 0																0000
WREG1	0002	工作寄存器 1																0000
WREG2	0004	工作寄存器 2																0000
WREG3	0006	工作寄存器 3																0000
WREG4	0008	工作寄存器 4																0000
WREG5	000A	工作寄存器 5																0000
WREG6	000C	工作寄存器 6																0000
WREG7	000E	工作寄存器 7																0000
WREG8	0010	工作寄存器 8																0000
WREG9	0012	工作寄存器 9																0000
WREG10	0014	工作寄存器 10																0000
WREG11	0016	工作寄存器 11																0000
WREG12	0018	工作寄存器 12																0000
WREG13	001A	工作寄存器 13																0000
WREG14	001C	工作寄存器 14																0000
WREG15	001E	工作寄存器 15																0800
SPLIM	0020	堆栈指针限制寄存器																xxxx
PCL	002E	程序计数器低位字寄存器																0000
PCH	0030	—	—	—	—	—	—	—	—	程序计数器高字节寄存器								0000
TBLPAG	0032	—	—	—	—	—	—	—	—	表页地址指针寄存器								0000
PSVPAG	0034	—	—	—	—	—	—	—	—	程序存储器可视性页地址指针寄存器								0000
RCOUNT	0036	Repeat 循环计数器寄存器																xxxx
DCOUNT	0038	DCOUNT<15:0>																xxxx
DOSTARTL	003A	DOSTARTL<15:1>															0	xxxx
DOSTARTH	003C	—	—	—	—	—	—	—	—	—	—	DOSTARTH<5:0>						00xx
DOENDL	003E	DOENDL<15:1>															0	xxxx
DOENDH	0040	—	—	—	—	—	—	—	—	—	—	DOENDH						00xx
SR	0042	OA	OB	SA	SB	OAB	SAB	DA	DC	IPL2	IPL1	IPL0	RA	N	OV	Z	C	0000
CORCON	0044	—	—	—	US	EDT	DL<2:0>			SATA	SATB	SATDW	ACCSAT	IPL3	PSV	RND	IF	0000
MODCON	0046	XMODEN	YMODEN	—	—	BWM<3:0>				YWM<3:0>				XWM<3:0>				0000
XMODSRT	0048	XS<15:1>															0	xxxx
XMODEND	004A	XE<15:1>															1	xxxx
YMODSRT	004C	YS<15:1>															0	xxxx
YMODEND	004E	YE<15:1>															1	xxxx
XBREV	0050	BREN	XB<14:0>															xxxx
DISICNT	0052	—	—	禁止中断计数器寄存器														xxxx
BSRAM	0750	—	—	—	—	—	—	—	—	—	—	—	—	—	IW_BSR	IR_BSR	RL_BSR	0000
SSRAM	0752	—	—	—	—	—	—	—	—	—	—	—	—	—	IW_SSR	IR_SSR	RL_SSR	0000

图注：　x = 复位时的未知值，— = 未实现 （读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-2：　　电平变化通知寄存器映射

SFR 名称	SFR 地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
CNEN1	0060	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	CN10IE	CN9IE	CN8IE	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE	CN1IE	CN0IE	0000
CNEN2	0062	—	—	—	—	—	—	—	—	CN23IE	CN22IE	CN21IE	CN20IE	CN19IE	CN18IE	CN17IE	CN16IE	0000
CNPU1	0068	CN15PUE	CN14PUE	CN13PUE	CN12PUE	CN11PUE	CN10PUE	CN9PUE	CN8PUE	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000
CNPU2	006A	—	—	—	—	—	—	—	—	CN23PUE	CN22PUE	CN21PUE	CN20PUE	CN19PUE	CN18PUE	CN17PUE	CN16PUE	0000

图注：　　x = 复位时的未知值，— = 未实现（读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-3：　中断控制器寄存器映射

SFR 名称	SFR 地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
INTCON1	0080	NSTDIS	OVAERR	OVBERR	COVAERR	COVBERR	OVATE	OVBTE	COVTE	SFTACERR	DIV0ERR	DMACERR	MATHERR	ADDRERR	STKERR	OSCFAIL	—	0000
INTCON2	0082	ALTIVT	DISI	—	—	—	—	—	—	—	—	—	INT4EP	INT3EP	INT2EP	INT1EP	INT0EP	0000
IFS0	0084	—	DMA1IF	AD1IF	U1TXIF	U1RXIF	SPI1IF	SPI1EIF	T3IF	T2IF	OC2IF	IC2IF	DMA0IF	T1IF	OC1IF	IC1IF	INT0IF	0000
IFS1	0086	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	DMA2IF	IC8IF	IC7IF	AD2IF	INT1IF	CNIF	—	MI2C1IF	SI2C1IF	0000
IFS2	0088	T6IF	DMA4IF		OC8IF	OC7IF	OC6IF	OC5IF	IC6IF	IC5IF	IC4IF	IC3IF	DMA3IF	C1IF	C1RXIF	SPI2IF	SPI2EIF	0000
IFS3	008A		—	DMA5IF	DCIIF	DCIEIF			C2IF	C2RXIF	INT4IF	INT3IF	T9IF	T8IF	MI2C2IF	SI2C2IF	T7IF	0000
IFS4	008C	—	—	—	—	—	—	—	—	C2TXIF	C1TXIF	DMA7IF	DMA6IF	—	U2EIF	U1EIF		0000
IEC0	0094	—	DMA1IE	AD1IE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE	T2IE	OC2IE	IC2IE	DMA0IE	T1IE	OC1IE	IC1IE	INT0IE	0000
IEC1	0096	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	DMA2IE	IC8IE	IC7IE	AD2IE	INT1IE	CNIE	—	MI2C1IE	SI2C1IE	0000
IEC2	0098	T6IE	DMA4IE	—	OC8IE	OC7IE	OC6IE	OC5IE	IC6IE	IC5IE	IC4IE	IC3IE	DMA3IE	C1IE	C1RXIE	SPI2IE	SPI2EIE	0000
IEC3	009A		—	DMA5IE	DCIIE	DCIEIE			C2IE	C2RXIE	INT4IE	INT3IE	T9IE	T8IE	MI2C2IE	SI2C2IE	T7IE	0000
IEC4	009C	—	—	—	—	—	—	—	—	C2TXIE	C1TXIE	DMA7IE	DMA6IE	—	U2EIE	U1EIE		0000
IPC0	00A4	—	T1IP<2:0>			—	OC1IP<2:0>			—	IC1IP<2:0>			—	INT0IP<2:0>			4444
IPC1	00A6	—	T2IP<2:0>			—	OC2IP<2:0>			—	IC2IP<2:0>			—	DMA0IP<2:0>			4444
IPC2	00A8	—	U1RXIP<2:0>			—	SPI1IP<2:0>			—	SPI1EIP<2:0>			—	T3IP<2:0>			4444
IPC3	00AA	—	—	—	—	—	DMA1IP<2:0>			—	AD1IP<2:0>			—	U1TXIP<2:0>			4444
IPC4	00AC	—	CNIP<2:0>			—	—	—	—	—	MI2C1IP<2:0>			—	SI2C1IP<2:0>			4444
IPC5	00AE	—	IC8IP<2:0>			—	IC7IP<2:0>			—	AD2IP<2:0>			—	INT1IP<2:0>			4444
IPC6	00B0	—	T4IP<2:0>			—	OC4IP<2:0>			—	OC3IP<2:0>			—	DMA2IP<2:0>			4444
IPC7	00B2	—	U2TXIP<2:0>			—	U2RXIP<2:0>			—	INT2IP<2:0>			—	T5IP<2:0>			4444
IPC8	00B4	—	C1IP<2:0>			—	C1RXIP<2:0>			—	SPI2IP<2:0>			—	SPI2EIP<2:0>			4444
IPC9	00B6	—	IC5IP<2:0>			—	IC4IP<2:0>			—	IC3IP<2:0>			—	DMA3IP<2:0>			4444
IPC10	00B8	—	OC7IP<2:0>			—	OC6IP<2:0>			—	OC5IP<2:0>			—	IC6IP<2:0>			4444
IPC11	00BA	—	T6IP<2:0>			—	DMA4IP<2:0>			—	—	—	—	—	OC8IP<2:0>			4444
IPC12	00BC	—	T8IP<2:0>			—	MI2C2IP<2:0>			—	SI2C2IP<2:0>			—	T7IP<2:0>			4444
IPC13	00BE	—	C2RXIP<2:0>			—	INT4IP<2:0>			—	INT3IP<2:0>			—	T9IP<2:0>			4444
IPC14	00C0	—	DCIEIP<2:0>			—				—				—	C2IP<2:0>			4444
IPC15	00C2	—				—	—	—	—	—	DMA5IP<2:0>			—	DCIIP<2:0>			4444
IPC16	00C4	—	—	—	—	—	U2EIP<2:0>			—	U1EIP<2:0>			—				4444
IPC17	00C6	—	C2TXIP<2:0>			—	C1TXIP<2:0>			—	DMA7IP<2:0>			—	DMA6IP<2:0>			4444
INTTREG	00E0	—	—	—	—	ILR<3:0>					VECNUM<6:0>							0000

图注：　x = 复位时的未知值，— = 未实现 （读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-4：　定时器寄存器映射

SFR 名称	SFR 地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
TMR1	0100	Timer1 寄存器																xxxx
PR1	0102	周期寄存器 1																FFFF
T1CON	0104	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>		—	TSYNC	TCS	—	0000
TMR2	0106	Timer2 寄存器																xxxx
TMR3HLD	0108	Timer3 保持寄存器（仅适用于 32 位定时器操作）																xxxx
TMR3	010A	Timer3 寄存器																xxxx
PR2	010C	周期寄存器 2																FFFF
PR3	010E	周期寄存器 3																FFFF
T2CON	0110	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>		T32	—	TCS	—	0000
T3CON	0112	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>		—	—	TCS	—	0000
TMR4	0114	Timer4 寄存器																xxxx
TMR5HLD	0116	Timer5 保持寄存器（仅适用于 32 位定时器操作）																xxxx
TMR5	0118	Timer5 寄存器																xxxx
PR4	011A	周期寄存器 4																FFFF
PR5	011C	周期寄存器 5																FFFF
T4CON	011E	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>		T32	—	TCS	—	0000
T5CON	0120	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>		—	—	TCS	—	0000
TMR6	0122	Timer6 寄存器																xxxx
TMR7HLD	0124	Timer7 保持寄存器（仅适用于 32 位定时器操作）																xxxx
TMR7	0126	Timer7 寄存器																xxxx
PR6	0128	周期寄存器 6																FFFF
PR7	012A	周期寄存器 7																FFFF
T6CON	012C	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>		T32	—	TCS	—	0000
T7CON	012E	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>		—	—	TCS	—	0000
TMR8	0130	Timer8 寄存器																xxxx
TMR9HLD	0132	Timer9 保持寄存器（仅适用于 32 位定时器操作）																xxxx
TMR9	0134	Timer9 寄存器																xxxx
PR8	0136	周期寄存器 8																FFFF
PR9	0138	周期寄存器 9																FFFF
T8CON	013A	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>		T32	—	TCS	—	0000
T9CON	013C	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>		—	—	TCS	—	0000

图注：　x = 复位时的未知值，— = 未实现（读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-5：　　输入捕捉寄存器映射

SFR 名称	SFR 地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
IC1BUF	0140	输入捕捉 1 寄存器																xxxx
IC1CON	0142	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000
IC2BUF	0144	输入捕捉 2 寄存器																xxxx
IC2CON	0146	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000
IC3BUF	0148	输入捕捉 3 寄存器																xxxx
IC3CON	014A	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000
IC4BUF	014C	输入捕捉 4 寄存器																xxxx
IC4CON	014E	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000
IC5BUF	0150	输入捕捉 5 寄存器																xxxx
IC5CON	0152	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000
IC6BUF	0154	输入捕捉 6 寄存器																xxxx
IC6CON	0156	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000
IC7BUF	0158	输入捕捉 7 寄存器																xxxx
IC7CON	015A	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000
IC8BUF	015C	输入捕捉 8 寄存器																xxxx
IC8CON	015E	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000

图注：　x = 复位时的未知值，— = 未实现 （读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-6：　　输出比较寄存器映射

SFR 名称	SFR 地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
OC1RS	0180	输出比较 1 辅助寄存器																xxxx
OC1R	0182	输出比较 1 寄存器																xxxx
OC1CON	0184	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>			0000
OC2RS	0186	输出比较 2 辅助寄存器																xxxx
OC2R	0188	输出比较 2 寄存器																xxxx
OC2CON	018A	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>			0000
OC3RS	018C	输出比较 3 辅助寄存器																xxxx
OC3R	018E	输出比较 3 寄存器																xxxx
OC3CON	0190	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>			0000
OC4RS	0192	输出比较 4 辅助寄存器																xxxx
OC4R	0194	输出比较 4 寄存器																xxxx
OC4CON	0196	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>			0000
OC5RS	0198	输出比较 5 辅助寄存器																xxxx
OC5R	019A	输出比较 5 寄存器																xxxx
OC5CON	019C	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>			0000
OC6RS	019E	输出比较 6 辅助寄存器																xxxx
OC6R	01A0	输出比较 6 寄存器																xxxx
OC6CON	01A2	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>			0000
OC7RS	01A4	输出比较 7 辅助寄存器																xxxx
OC7R	01A6	输出比较 7 寄存器																xxxx
OC7CON	01A8	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>			0000
OC8RS	01AA	输出比较 8 辅助寄存器																xxxx
OC8R	01AC	输出比较 8 寄存器																xxxx
OC8CON	01AE	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>			0000

图注： x = 复位时的未知值， — = 未实现 （读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-7： I2C1 寄存器映射

SFR 名称	SFR 地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
I2C1RCV	0200	—	—	—	—	—	—	—	—	接收寄存器								0000
I2C1TRN	0202	—	—	—	—	—	—	—	—	发送寄存器								00FF
I2C1BRG	0204	—	—	—	—	—	—	—	波特率发生器寄存器									0000
I2C1CON	0206	I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
I2C1STAT	0208	ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D_A	P	S	R_W	RBF	TBF	0000
I2C1ADD	020A	—	—	—	—	—	—	地址寄存器										0000
I2C1MSK	020C	—	—	—	—	—	—	地址掩码寄存器										0000

图注： x = 复位时的未知值， — = 未实现 （读为 0）。复位值以十六进制显示。

表 3-8： I2C2 寄存器映射

SFR 名称	SFR 地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
I2C2RCV	0210	—	—	—	—	—	—	—	—	接收寄存器								0000
I2C2TRN	0212	—	—	—	—	—	—	—	—	发送寄存器								00FF
I2C2BRG	0214	—	—	—	—	—	—	—	波特率发生器寄存器									0000
I2C2CON	0216	I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
I2C2STAT	0218	ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D_A	P	S	R_W	RBF	TBF	0000
I2C2ADD	021A	—	—	—	—	—	—	地址寄存器										0000
I2C2MSK	021C	—	—	—	—	—	—	地址掩码寄存器										0000

图注： x = 复位时的未知值， — = 未实现 （读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-9： UART1 寄存器映射

SFR 名称	SFR 地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
U1MODE	0220	UARTEN	—	USIDL	IREN	RTSMD	—	UEN1	UEN0	WAKE	LPBACK	ABAUD	URXINV	BRGH	PDSEL<1:0>		STSEL	0000
U1STA	0222	UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL<1:0>		ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110
U1TXREG	0224	—	—	—	—	—	—	—	UART 发送寄存器									xxxx
U1RXREG	0226	—	—	—	—	—	—	—	UART 接收寄存器									0000
U1BRG	0228	波特率发生器预分频器																0000

图注： x = 复位时的未知值，— = 未实现（读为 0）。复位值以十六进制显示。

表 3-10： UART2 寄存器映射

SFR 名称	SFR 地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
U2MODE	0230	UARTEN	—	USIDL	IREN	RTSMD	—	UEN1	UEN0	WAKE	LPBACK	ABAUD	URXINV	BRGH	PDSEL<1:0>		STSEL	0000
U2STA	0232	UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL<1:0>		ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110
U2TXREG	0234	—	—	—	—	—	—	—	UART 发送寄存器									xxxx
U2RXREG	0236	—	—	—	—	—	—	—	UART 接收寄存器									0000
U2BRG	0238	波特率发生器预分频器																0000

图注： x = 复位时的未知值，— = 未实现（读为 0）。复位值以十六进制显示。

表 3-11： SPI1 寄存器映射

SFR 名称	SFR 地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
SPI1STAT	0240	SPIEN	—	SPISIDL	—	—	—	—	—	—	SPIROV	—	—	—	—	SPITBF	SPIRBF	0000
SPI1CON1	0242	—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE<2:0>			PPRE<1:0>		0000
SPI1CON2	0244	FRMEN	SPIFSD	FRMPOL	—	—	—	—	—	—	—	—	—	—	—	FRMDLY	—	0000
SPI1BUF	0248	SPI1 发送和接收缓冲寄存器																0000

图注： x = 复位时的未知值，— = 未实现（读为 0）。复位值以十六进制显示。

表 3-12： SPI2 寄存器映射

SFR 名称	SFR 地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
SPI2STAT	0260	SPIEN	—	SPISIDL	—	—	—	—	—	—	SPIROV	—	—	—	—	SPITBF	SPIRBF	0000
SPI2CON1	0262	—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE<2:0>			PPRE<1:0>		0000
SPI2CON2	0264	FRMEN	SPIFSD	FRMPOL	—	—	—	—	—	—	—	—	—	—	—	FRMDLY	—	0000
SPI2BUF	0268	SPI2 发送和接收缓冲寄存器																0000

图注： x = 复位时的未知值，— = 未实现（读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-13： ADC1 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
ADC1BUF0	0300	ADC 数据缓冲器 0																xxxx
AD1CON1	0320	ADON	—	ADSIDL	ADDMABM	—	AD12B	FORM<1:0>		SSRC<2:0>			—	SIMSAM	ASAM	SAMP	DONE	0000
AD1CON2	0322	VCFG<2:0>			—	—	CSCNA	CHPS<1:0>		BUFS	—	SMPI<3:0>				BUFM	ALTS	0000
AD1CON3	0324	ADRC	—	—	SAMC<4:0>					—	—	ADCS<5:0>						0000
AD1CHS123	0326	—	—	—	—	—	CH123NB<1:0>		CH123SB	—	—	—	—	—	CH123NA<1:0>		CH123SA	0000
AD1CHS0	0328	CH0NB	—	—	CH0SB<4:0>					CH0NA	—	—	CH0SA<4:0>					0000
AD1PCFGH	032A	PCFG31	PCFG30	PCFG29	PCFG28	PCFG27	PCFG26	PCFG25	PCFG24	PCFG23	PCFG22	PCFG21	PCFG20	PCFG19	PCFG18	PCFG17	PCFG16	0000
AD1PCFGL	032C	PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0	0000
AD1CSSH	032E	CSS31	CSS30	CSS29	CSS28	CSS27	CSS26	CSS25	CSS24	CSS23	CSS22	CSS21	CSS20	CSS19	CSS18	CSS17	CSS16	0000
AD1CSSL	0330	CSS15	CSS14	CSS13	CSS12	CSS11	CSS10	CSS9	CSS8	CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0	0000
AD1CON4	0332	—	—	—	—	—	—	—	—	—	—	—	—	—	DMABL<2:0>			0000

图注： x = 复位时的未知值， — = 未实现 （读为 0）。复位值以十六进制显示。

表 3-14： ADC2 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
ADC2BUF0	0340	ADC 数据缓冲器 0																xxxx
AD2CON1	0360	ADON	—	ADSIDL	ADDMABM	—	AD12B	FORM<1:0>		SSRC<2:0>			—	SIMSAM	ASAM	SAMP	DONE	0000
AD2CON2	0362	VCFG<2:0>			—	—	CSCNA	CHPS<1:0>		BUFS	—	SMPI<3:0>				BUFM	ALTS	0000
AD2CON3	0364	ADRC	—	—	SAMC<4:0>					—	—	ADCS<5:0>						0000
AD2CHS123	0366	—	—	—	—	—	CH123NB<1:0>		CH123SB	—	—	—	—	—	CH123NA<1:0>		CH123SA	0000
AD2CHS0	0368	CH0NB	—	—	—	CH0SB<3:0>				CH0NA	—	—	—	CH0SA<3:0>				0000
保留	036A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
AD2PCFGL	036C	PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0	0000
保留	036E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
AD2CSSL	0370	CSS15	CSS14	CSS13	CSS12	CSS11	CSS10	CSS9	CSS8	CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0	0000
AD2CON4	0372	—	—	—	—	—	—	—	—	—	—	—	—	—	DMABL<2:0>			0000

图注： x = 复位时的未知值， — = 未实现 （读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-15：　DMA 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
DMA0CON	0380	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA0REQ	0382	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA0STA	0384	STA<15:0>																0000
DMA0STB	0386	STB<15:0>																0000
DMA0PAD	0388	PAD<15:0>																0000
DMA0CNT	038A	—	—	—	—	—	—	CNT<9:0>										0000
DMA1CON	038C	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA1REQ	038E	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA1STA	0390	STA<15:0>																0000
DMA1STB	0392	STB<15:0>																0000
DMA1PAD	0394	PAD<15:0>																0000
DMA1CNT	0396	—	—	—	—	—	—	CNT<9:0>										0000
DMA2CON	0398	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA2REQ	039A	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA2STA	039C	STA<15:0>																0000
DMA2STB	039E	STB<15:0>																0000
DMA2PAD	03A0	PAD<15:0>																0000
DMA2CNT	03A2	—	—	—	—	—	—	CNT<9:0>										0000
DMA3CON	03A4	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA3REQ	03A6	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA3STA	03A8	STA<15:0>																0000
DMA3STB	03AA	STB<15:0>																0000
DMA3PAD	03AC	PAD<15:0>																0000
DMA3CNT	03AE	—	—	—	—	—	—	CNT<9:0>										0000
DMA4CON	03B0	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA4REQ	03B2	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA4STA	03B4	STA<15:0>																0000
DMA4STB	03B6	STB<15:0>																0000
DMA4PAD	03B8	PAD<15:0>																0000
DMA4CNT	03BA	—	—	—	—	—	—	CNT<9:0>										0000
DMA5CON	03BC	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA5REQ	03BE	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA5STA	03C0	STA<15:0>																0000
DMA5STB	03C2	STB<15:0>																0000
DMA5PAD	03C4	PAD<15:0>																0000

图注：　— = 未实现 （读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-15： DMA 寄存器映射 （续）

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
DMA5CNT	03C6	—	—	—	—	—	—	CNT<9:0>										0000
DMA6CON	03C8	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA6REQ	03CA	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA6STA	03CC	STA<15:0>																0000
DMA6STB	03CE	STB<15:0>																0000
DMA6PAD	03D0	PAD<15:0>																0000
DMA6CNT	03D2	—	—	—	—	—	—	CNT<9:0>										0000
DMA7CON	03D4	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA7REQ	03D6	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA7STA	03D8	STA<15:0>																0000
DMA7STB	03DA	STB<15:0>																0000
DMA7PAD	03DC	PAD<15:0>																0000
DMA7CNT	03DE	—	—	—	—	—	—	CNT<9:0>										0000
DMACS0	03E0	PWCOL7	PWCOL6	PWCOL5	PWCOL4	PWCOL3	PWCOL2	PWCOL1	PWCOL0	XWCOL7	XWCOL6	XWCOL5	XWCOL4	XWCOL3	XWCOL2	XWCOL1	XWCOL0	0000
DMACS1	03E2	—	—	—	—	LSTCH<3:0>				PPST7	PPST6	PPST5	PPST4	PPST3	PPST2	PPST1	PPST0	0000
DSADR	03E4	DSADR<15:0>																0000

图注： — = 未实现 （读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-16：　当 C1CTRL1.WIN = 0 或 1 时的 ECAN1 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
C1CTRL1	0400	—	—	CSIDL	ABAT	CANCKS	REQOP<2:0>			OPMODE<2:0>			—	CANCAP	—	—	WIN	0480
C1CTRL2	0402	—	—	—	—	—	—	—	—	—	—	—	DNCNT<4:0>					0000
C1VEC	0404	—	—	—	FILHIT<4:0>					—	ICODE<6:0>							0000
C1FCTRL	0406	DMABS<2:0>			—	—	—	—	—	—	—	—	FSA<4:0>					0000
C1FIFO	0408	—	—	FBP<5:0>						—	—	FNRB<5:0>						0000
C1INTF	040A	—	—	TXBO	TXBP	RXBP	TXWAR	RXWAR	EWARN	IVRIF	WAKIF	ERRIF	—	FIFOIF	RBOVIF	RBIF	TBIF	0000
C1INTE	040C	—	—	—	—	—	—	—	—	IVRIE	WAKIE	ERRIE	—	FIFOIE	RBOVIE	RBIE	TBIE	0000
C1EC	040E	TERRCNT<7:0>								RERRCNT<7:0>								0000
C1CFG1	0410	—	—	—	—	—	—	—	—	SJW<1:0>		BRP<5:0>						0000
C1CFG2	0412	—	WAKFIL	—	—	—	SEG2PH<2:0>			SEG2PHTS	SAM	SEG1PH<2:0>			PRSEG<2:0>			0000
C1FEN1	0414	FLTEN15	FLTEN14	FLTEN13	FLTEN12	FLTEN11	FLTEN10	FLTEN9	FLTEN8	FLTEN7	FLTEN6	FLTEN5	FLTEN4	FLTEN3	FLTEN2	FLTEN1	FLTEN0	0000
C1FMSKSEL1	0418	F7MSK<1:0>		F6MSK<1:0>		F5MSK<1:0>		F4MSK<1:0>		F3MSK<1:0>		F2MSK<1:0>		F1MSK<1:0>		F0MSK<1:0>		0000
C1FMSKSEL2	041A	F15MSK<1:0>		F14MSK<1:0>		F13MSK<1:0>		F12MSK<1:0>		F11MSK<1:0>		F10MSK<1:0>		F9MSK<1:0>		F8MSK<1:0>		0000

图注：　— = 未实现 （读为 0）。复位值以十六进制显示。

表 3-17：　当 C1CTRL1.WIN = 0 时的 ECAN1 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
	0400-041E	请参见当 WIN = x 时的定义																
C1RXFUL1	0420	RXFUL15	RXFUL14	RXFUL13	RXFUL12	RXFUL11	RXFUL10	RXFUL9	RXFUL8	RXFUL7	RXFUL6	RXFUL5	RXFUL4	RXFUL3	RXFUL2	RXFUL1	RXFUL0	0000
C1RXFUL2	0422	RXFUL31	RXFUL30	RXFUL29	RXFUL28	RXFUL27	RXFUL26	RXFUL25	RXFUL24	RXFUL23	RXFUL22	RXFUL21	RXFUL20	RXFUL19	RXFUL18	RXFUL17	RXFUL16	0000
C1RXOVF1	0428	RXOVF15	RXOVF14	RXOVF13	RXOVF12	RXOVF11	RXOVF10	RXOVF9	RXOVF8	RXOVF7	RXOVF6	RXOVF5	RXOVF4	RXOVF3	RXOVF2	RXOVF1	RXOVF0	0000
C1RXOVF2	042A	RXOVF31	RXOVF30	RXOVF29	RXOVF28	RXOVF27	RXOVF26	RXOVF25	RXOVF24	RXOVF23	RXOVF22	RXOVF21	RXOVF20	RXOVF19	RXOVF18	RXOVF17	RXOVF16	0000
C1TR01CON	0430	TXEN1	TXABT1	TXLARB1	TXERR1	TXREQ1	RTREN1	TX1PRI<1:0>		TXEN0	TXABAT0	TXLARB0	TXERR0	TXREQ0	RTREN0	TX0PRI<1:0>		0000
C1TR23CON	0432	TXEN3	TXABT3	TXLARB3	TXERR3	TXREQ3	RTREN3	TX3PRI<1:0>		TXEN2	TXABAT2	TXLARB2	TXERR2	TXREQ2	RTREN2	TX2PRI<1:0>		0000
C1TR45CON	0434	TXEN5	TXABT5	TXLARB5	TXERR5	TXREQ5	RTREN5	TX5PRI<1:0>		TXEN4	TXABAT4	TXLARB4	TXERR4	TXREQ4	RTREN4	TX4PRI<1:0>		0000
C1TR67CON	0436	TXEN7	TXABT7	TXLARB7	TXERR7	TXREQ7	RTREN7	TX7PRI<1:0>		TXEN6	TXABAT6	TXLARB6	TXERR6	TXREQ6	RTREN6	TX6PRI<1:0>		xxxx
C1RXD	0440	接收到的数据字																xxxx
C1TXD	0442	待发送的数据字																xxxx

图注：　x = 复位时的未知值，— = 未实现 （读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-18：　当 C1CTRL1.WIN = 1 时的 ECAN1 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
	0400-041E	请参见当 WIN = x 时的定义																
C1BUFPNT1	0420	F3BP<3:0>				F2BP<3:0>				F1BP<3:0>				F0BP<3:0>				0000
C1BUFPNT2	0422	F7BP<3:0>				F6BP<3:0>				F5BP<3:0>				F4BP<3:0>				0000
C1BUFPNT3	0424	F11BP<3:0>				F10BP<3:0>				F9BP<3:0>				F8BP<3:0>				0000
C1BUFPNT4	0426	F15BP<3:0>				F14BP<3:0>				F13BP<3:0>				F12BP<3:0>				0000
C1RXM0SID	0430	SID<10:3>								SID<2:0>			—	MIDE	—	EID<17:16>		xxxx
C1RXM0EID	0432	EID<15:8>								EID<7:0>								xxxx
C1RXM1SID	0434	SID<10:3>								SID<2:0>			—	MIDE	—	EID<17:16>		xxxx
C1RXM1EID	0436	EID<15:8>								EID<7:0>								xxxx
C1RXM2SID	0438	SID<10:3>								SID<2:0>			—	MIDE	—	EID<17:16>		xxxx
C1RXM2EID	043A	EID<15:8>								EID<7:0>								xxxx
C1RXF0SID	0440	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF0EID	0442	EID<15:8>								EID<7:0>								xxxx
C1RXF1SID	0444	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF1EID	0446	EID<15:8>								EID<7:0>								xxxx
C1RXF2SID	0448	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF2EID	044A	EID<15:8>								EID<7:0>								xxxx
C1RXF3SID	044C	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF3EID	044E	EID<15:8>								EID<7:0>								xxxx
C1RXF4SID	0450	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF4EID	0452	EID<15:8>								EID<7:0>								xxxx
C1RXF5SID	0454	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF5EID	0456	EID<15:8>								EID<7:0>								xxxx
C1RXF6SID	0458	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF6EID	045A	EID<15:8>								EID<7:0>								xxxx
C1RXF7SID	045C	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF7EID	045E	EID<15:8>								EID<7:0>								xxxx
C1RXF8SID	0460	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF8EID	0462	EID<15:8>								EID<7:0>								xxxx
C1RXF9SID	0464	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF9EID	0466	EID<15:8>								EID<7:0>								xxxx
C1RXF10SID	0468	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF10EID	046A	EID<15:8>								EID<7:0>								xxxx

图注：　x = 复位时的未知值，— = 未实现（读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-18：　　当 C1CTRL1.WIN = 1 时的 ECAN1 寄存器映射 （续）

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
C1RXF11SID	046C	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF11EID	046E	EID<15:8>								EID<7:0>								xxxx
C1RXF12SID	0470	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF12EID	0472	EID<15:8>								EID<7:0>								xxxx
C1RXF13SID	0474	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF13EID	0476	EID<15:8>								EID<7:0>								xxxx
C1RXF14SID	0478	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF14EID	047A	EID<15:8>								EID<7:0>								xxxx
C1RXF15SID	047C	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C1RXF15EID	047E	EID<15:8>								EID<7:0>								xxxx

图注：　　x = 复位时的未知值， — = 未实现 （读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-19：　当 C2CTRL1.WIN = 0 或 1 时的 ECAN2 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
C2CTRL1	0500	—	—	CSIDL	ABAT	CANCKS	REQOP<2:0>			OPMODE<2:0>			—	CANCAP	—	—	WIN	0480
C2CTRL2	0502	—	—	—	—	—	—	—	—	—	—	—	DNCNT<4:0>					0000
C2VEC	0504	—	—	—	FILHIT<4:0>					—	ICODE<6:0>							0000
C2FCTRL	0506	DMABS<2:0>			—	—	—	—	—	—	—	—	FSA<4:0>					0000
C2FIFO	0508	—	—	FBP<5:0>						—	—	FNRB<5:0>						0000
C2INTF	050A	—	—	TXBO	TXBP	RXBP	TXWAR	RXWAR	EWARN	IVRIF	WAKIF	ERRIF	—	FIFOIF	RBOVIF	RBIF	TBIF	0000
C2INTE	050C	—	—	—	—	—	—	—	—	IVRIE	WAKIE	ERRIE	—	FIFOIE	RBOVIE	RBIE	TBIE	0000
C2EC	050E	TERRCNT<7:0>								RERRCNT<7:0>								0000
C2CFG1	0510	—	—	—	—	—	—	—	—	SJW<1:0>		BRP<5:0>						0000
C2CFG2	0512	—	WAKFIL	—	—	—	SEG2PH<2:0>			SEG2PHTS	SAM	SEG1PH<2:0>			PRSEG<2:0>			0000
C2FEN1	0514	FLTEN15	FLTEN14	FLTEN13	FLTEN12	FLTEN11	FLTEN10	FLTEN9	FLTEN8	FLTEN7	FLTEN6	FLTEN5	FLTEN4	FLTEN3	FLTEN2	FLTEN1	FLTEN0	0000
C2FMSKSEL1	0518	F7MSK<1:0>		F6MSK<1:0>		F5MSK<1:0>		F4MSK<1:0>		F3MSK<1:0>		F2MSK<1:0>		F1MSK<1:0>		F0MSK<1:0>		0000
C2FMSKSEL2	051A	F15MSK<1:0>		F14MSK<1:0>		F13MSK<1:0>		F12MSK<1:0>		F11MSK<1:0>		F10MSK<1:0>		F9MSK<1:0>		F8MSK<1:0>		0000

图注：　— = 未实现 （读为 0）。复位值以十六进制显示。

表 3-20：　当 C2CTRL1.WIN = 0 时的 ECAN2 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
	0500-051E	请参见当 WIN = x 时的定义																
C2RXFUL1	0520	RXFUL15	RXFUL14	RXFUL13	RXFUL12	RXFUL11	RXFUL10	RXFUL9	RXFUL8	RXFUL7	RXFUL6	RXFUL5	RXFUL4	RXFUL3	RXFUL2	RXFUL1	RXFUL0	0000
C2RXFUL2	0522	RXFUL31	RXFUL30	RXFUL29	RXFUL28	RXFUL27	RXFUL26	RXFUL25	RXFUL24	RXFUL23	RXFUL22	RXFUL21	RXFUL20	RXFUL19	RXFUL18	RXFUL17	RXFUL16	0000
C2RXOVF1	0528	RXOVF15	RXOVF14	RXOVF13	RXOVF12	RXOVF11	RXOVF10	RXOVF09	RXOVF08	RXOVF7	RXOVF6	RXOVF5	RXOVF4	RXOVF3	RXOVF2	RXOVF1	RXOVF0	0000
C2RXOVF2	052A	RXOVF31	RXOVF30	RXOVF29	RXOVF28	RXOVF27	RXOVF26	RXOVF25	RXOVF24	RXOVF23	RXOVF22	RXOVF21	RXOVF20	RXOVF19	RXOVF18	RXOVF17	RXOVF16	0000
C2TR01CON	0530	TXEN1	TX ABAT1	TX LARB1	TX ERR1	TX REQ1	RTREN1	TX1PRI<1:0>		TXEN0	TX ABAT0	TX LARB0	TX ERR0	TX REQ0	RTREN0	TX0PRI<1:0>		0000
C2TR23CON	0532	TXEN3	TX ABAT3	TX LARB3	TX ERR3	TX REQ3	RTREN3	TX3PRI<1:0>		TXEN2	TX ABAT2	TX LARB2	TX ERR2	TX REQ2	RTREN2	TX2PRI<1:0>		0000
C2TR45CON	0534	TXEN5	TX ABAT5	TX LARB5	TX ERR5	TX REQ5	RTREN5	TX5PRI<1:0>		TXEN4	TX ABAT4	TX LARB4	TX ERR4	TX REQ4	RTREN4	TX4PRI<1:0>		0000
C2TR67CON	0536	TXEN7	TX ABAT7	TX LARB7	TX ERR7	TX REQ7	RTREN7	TX7PRI<1:0>		TXEN6	TX ABAT6	TX LARB6	TX ERR6	TX REQ6	RTREN6	TX6PRI<1:0>		xxxx
C2RXD	0540	接收到的数据字																xxxx
C2TXD	0542	待发送的数据字																xxxx

图注：　x = 复位时的未知值，— = 未实现 （读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-21：　当 C2CTRL1.WIN = 1 时的 ECAN2 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
	0500 - 051E	请参见当 WIN = x 时的定义																
C2BUFPNT1	0520	F3BP<3:0>				F2BP<3:0>				F1BP<3:0>				F0BP<3:0>				0000
C2BUFPNT2	0522	F7BP<3:0>				F6BP<3:0>				F5BP<3:0>				F4BP<3:0>				0000
C2BUFPNT3	0524	F11BP<3:0>				F10BP<3:0>				F9BP<3:0>				F8BP<3:0>				0000
C2BUFPNT4	0526	F15BP<3:0>				F14BP<3:0>				F13BP<3:0>				F12BP<3:0>				0000
C2RXM0SID	0530	SID<10:3>								SID<2:0>			—	MIDE	—	EID<17:16>		xxxx
C2RXM0EID	0532	EID<15:8>								EID<7:0>								xxxx
C2RXM1SID	0534	SID<10:3>								SID<2:0>			—	MIDE	—	EID<17:16>		xxxx
C2RXM1EID	0536	EID<15:8>								EID<7:0>								xxxx
C2RXM2SID	0538	SID<10:3>								SID<2:0>			—	MIDE	—	EID<17:16>		xxxx
C2RXM2EID	053A	EID<15:8>								EID<7:0>								xxxx
C2RXF0SID	0540	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C2RXF0EID	0542	EID<15:8>								EID<7:0>								xxxx
C2RXF1SID	0544	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C2RXF1EID	0546	EID<15:8>								EID<7:0>								xxxx
C2RXF2SID	0548	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C2RXF2EID	054A	EID<15:8>								EID<7:0>								xxxx
C2RXF3SID	054C	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C2RXF3EID	054E	EID<15:8>								EID<7:0>								xxxx
C2RXF4SID	0550	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C2RXF4EID	0552	EID<15:8>								EID<7:0>								xxxx
C2RXF5SID	0554	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C2RXF5EID	0556	EID<15:8>								EID<7:0>								xxxx
C2RXF6SID	0558	SID<10:3>								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C2RXF6EID	055A	EID<15:8>								EID<7:0>								xxxx
C2RXF7SID	055C	SID<10:3								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C2RXF7EID	055E	EID<15:8>								EID<7:0>								xxxx
C2RXF8SID	0560	SID<10:3								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C2RXF8EID	0562	EID<15:8>								EID<7:0>								xxxx
C2RXF9SID	0564	SID<10:3								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C2RXF9EID	0566	EID<15:8>								EID<7:0>								xxxx
C2RXF10SID	0568	SID<10:3								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx

图注：　x = 复位时的未知值，— = 未实现（读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-21：　当 C2CTRL1.WIN = 1 时的 ECAN2 寄存器映射 （续）

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
C2RXF10EID	056A	EID<15:8>								EID<7:0>								xxxx
C2RXF11SID	056C	SID<10:3								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C2RXF11EID	056E	EID<15:8>								EID<7:0>								xxxx
C2RXF12SID	0570	SID<10:3								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C2RXF12EID	0572	EID<15:8>								EID<7:0>								xxxx
C2RXF13SID	0574	SID<10:3								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C2RXF13EID	0576	EID<15:8>								EID<7:0>								xxxx
C2RXF14SID	0578	SID<10:3								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C2RXF14EID	057A	EID<15:8>								EID<7:0>								xxxx
C2RXF15SID	057C	SID<10:3								SID<2:0>			—	EXIDE	—	EID<17:16>		xxxx
C2RXF15EID	057E	EID<15:8>								EID<7:0>								xxxx

图注：　x = 复位时的未知值，— = 未实现 （读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

表 3-22：　DCI 寄存器映射

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
DCICON1	0280	DCIEN	—	DCISIDL	—	DLOOP	CSCKD	CSCKE	COFSD	UNFM	CSDOM	DJST	—	—	—	COFSM1	COFSM0	0000 0000 0000 0000
DCICON2	0282	—	—	—	—	BLEN1	BLEN0	—	COFSG<3:0>				—	WS<3:0>				0000 0000 0000 0000
DCICON3	0284	—	—	—	—	BCG<11:0>												0000 0000 0000 0000
DCISTAT	0286	—	—	—	—	SLOT3	SLOT2	SLOT1	SLOT0	—	—	—	—	ROV	RFUL	TUNF	TMPTY	0000 0000 0000 0000
TSCON	0288	TSE15	TSE14	TSE13	TSE12	TSE11	TSE10	TSE9	TSE8	TSE7	TSE6	TSE5	TSE4	TSE3	TSE2	TSE1	TSE0	0000 0000 0000 0000
RSCON	028C	RSE15	RSE14	RSE13	RSE12	RSE11	RSE10	RSE9	RSE8	RSE7	RSE6	RSE5	RSE4	RSE3	RSE2	RSE1	RSE0	0000 0000 0000 0000
RXBUF0	0290	接收缓冲器 #0 数据寄存器																0000 0000 0000 0000
RXBUF1	0292	接收缓冲器 #1 数据寄存器																0000 0000 0000 0000
RXBUF2	0294	接收缓冲器 #2 数据寄存器																0000 0000 0000 0000
RXBUF3	0296	接收缓冲器 #3 数据寄存器																0000 0000 0000 0000
TXBUF0	0298	发送缓冲器 #0 数据寄存器																0000 0000 0000 0000
TXBUF1	029A	发送缓冲器 #1 数据寄存器																0000 0000 0000 0000
TXBUF2	029C	发送缓冲器 #2 数据寄存器																0000 0000 0000 0000
TXBUF3	029E	发送缓冲器 #3 数据寄存器																0000 0000 0000 0000

图注：　— = 未实现（读为 0）。
注　1：　关于寄存器中各位的说明，请参见《dsPIC33F 系列参考手册》。

表 3-23：　PORTA 寄存器映射 (1)

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
TRISA	02C0	TRISA15	TRISA14	TRISA13	TRISA12	—	TRISA10	TRISA9	—	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	D6C0
PORTA	02C2	RA15	RA14	RA13	RA12	—	RA10	RA9	—	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx
LATA	02C4	LATA15	LATA14	LATA13	LATA12	—	LATA10	LATA9	—	LATA7	LATA6	LATA5	LATA4	LATA3	LATA2	LATA1	LATA0	xxxx
ODCA(2)	06C0	ODCA15	ODCA14	ODCA13	ODCA12	—	—	—	—	—	—	ODCA5	ODCA4	ODCA3	ODCA2	ODCA1	ODCA0	xxxx

图注：　x = 复位时的未知值，— = 未实现（读为 0）。PinHigh 器件的复位值以十六进制显示。
注　1：　实际的 I/O 端口引脚设置随器件变化。请参见相应的引脚图。

表 3-24：　PORTB 寄存器映射 (1)

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
TRISB	02C6	TRISB15	TRISB14	TRISB13	TRISB12	TRISB11	TRISB10	TRISB9	TRISB8	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	FFFF
PORTB	02C8	RB15	RB14	RB13	RB12	RB11	RB10	RB9	RB8	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx
LATB	02CA	LATB15	LATB14	LATB13	LATB12	LATB11	LATB10	LATB9	LATB8	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	xxxx

图注：　x = 复位时的未知值，— = 未实现（读为 0）。PinHigh 器件的复位值以十六进制显示。
注　1：　实际的 I/O 端口引脚设置随器件变化。请参见相应的引脚图。

© 2007 Microchip Technology Inc.

表 3-25： PORTC 寄存器映射 (1)

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
TRISC	02CC	TRISC15	TRISC14	TRISC13	TRISC12	—	—	—	—	—	—	—	TRISC4	TRISC3	TRISC2	TRISC1	—	F01E
PORTC	02CE	RC15	RC14	RC13	RC12	—	—	—	—	—	—	—	RC4	RC3	RC2	RC1	—	xxxx
LATC	02D0	LATC15	LATC14	LATC13	LATC12	—	—	—	—	—	—	—	LATC4	LATC3	LATC2	LATC1	—	xxxx

图注： x = 复位时的未知值， — = 未实现 （读为 0）。 PinHigh 器件的复位值以十六进制显示。
注 1： 实际的 I/O 端口引脚设置随器件变化。请参见相应的引脚图。

表 3-26： PORTD 寄存器映射 (1)

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
TRISD	02D2	TRISD15	TRISD14	TRISD13	TRISD12	TRISD11	TRISD10	TRISD9	TRISD8	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	FFFF
PORTD	02D4	RD15	RD14	RD13	RD12	RD11	RD10	RD9	RD8	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx
LATD	02D6	LATD15	LATD14	LATD13	LATD12	LATD11	LATD10	LATD9	LATD8	LATD7	LATD6	LATD5	LATD4	LATD3	LATD2	LATD1	LATD0	xxxx
ODCD	06D2	ODCD15	ODCD14	ODCD13	ODCD12	ODCD11	ODCD10	ODCD9	ODCD8	ODCD7	ODCD6	ODCD5	ODCD4	ODCD3	ODCD2	ODCD1	ODCD0	xxxx

图注： x = 复位时的未知值， — = 未实现 （读为 0）。 PinHigh 器件的复位值以十六进制显示。
注 1： 实际的 I/O 端口引脚设置随器件变化。请参见相应的引脚图。

表 3-27： PORTE 寄存器映射 (1)

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
TRISE	02D8	—	—	—	—	—	—	—	—	TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	TRISE2	TRISE1	TRISE0	03FF
PORTE	02DA	—	—	—	—	—	—	—	—	RE7	RE6	RE5	RE4	RE3	RE2	RE1	RE0	xxxx
LATE	02DC	—	—	—	—	—	—	—	—	LATE7	LATE6	LATE5	LATE4	LATE3	LATE2	LATE1	LATE0	xxxx

图注： x = 复位时的未知值， — = 未实现 （读为 0）。 PinHigh 器件的复位值以十六进制显示。
注 1： 实际的 I/O 端口引脚设置随器件变化。请参见相应的引脚图。

表 3-28： PORTF 寄存器映射 (1)

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
TRISF	02DE	—	—	TRISF13	TRISF12	—	—	—	TRISF8	TRISF7	TRISF6	TRISF5	TRISF4	TRISF3	TRISF2	TRISF1	TRISF0	31FF
PORTF	02E0	—	—	RF13	RF12	—	—	—	RF8	RF7	RF6	RF5	RF4	RF3	RF2	RF1	RF0	xxxx
LATF	02E2	—	—	LATF13	LATF12	—	—	—	LATF8	LATF7	LATF6	LATF5	LATF4	LATF3	LATF2	LATF1	LATF0	xxxx
ODCF	06DE	—	—	ODCF13	ODCF12	—	—	—	ODCF8	ODCF7	ODCF6	ODCF5	ODCF4	ODCF3	ODCF2	ODCF1	ODCF0	xxxx

图注： x = 复位时的未知值， — = 未实现 （读为 0）。 PinHigh 器件的复位值以十六进制显示。
注 1： 实际的 I/O 端口引脚设置随器件变化。请参见相应的引脚图。

© 2007 Microchip Technology Inc.

表 3-29： PORTG 寄存器映射 (1)

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
TRISG	02E4	TRISG15	TRISG14	TRISG13	TRISG12	—	—	TRISG9	TRISG8	TRISG7	TRISG6	—	—	TRISG3	TRISG2	TRISG1	TRISG0	F3CF
PORTG	02E6	RG15	RG14	RG13	RG12	—	—	RG9	RG8	RG7	RG6	—	—	RG3	RG2	RG1	RG0	xxxx
LATG	02E8	LATG15	LATG14	LATG13	LATG12	—	—	LATG9	LATG8	LATG7	LATG6	—	—	LATG3	LATG2	LATG1	LATG0	xxxx
ODCG	06E4	ODCG15	ODCG14	ODCG13	ODCG12	—	—	ODCG9	ODCG8	ODCG7	ODCG6	—	—	ODCG3	ODCG2	ODCG1	ODCG0	xxxx

图注： x = 复位时的未知值， — = 未实现 （读为 0）。 PinHigh 器件的复位值以十六进制显示。
注 1： 实际的 I/O 端口引脚设置随器件变化。请参见相应的引脚图。

表 3-30： 系统控制寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
RCON	0740	TRAPR	IOPUWR	—	—	—	—	—	VREGS	EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE	BOR	POR	xxxx(1)
OSCCON	0742	—	COSC<2:0>			—	NOSC<2:0>			CLKLOCK	—	LOCK	—	CF	—	LPOSCEN	OSWEN	0300(2)
CLKDIV	0744	ROI	DOZE<2:0>			DOZEN	FRCDIV<2:0>			PLLPOST<1:0>		—	PLLPRE<4::0>					0040
PLLFBD	0746	—	—	—	—	—	—	—	PLLDIV<8:0>									0030
OSCTUN	0748	—	—	—	—	—	—	—	—	—	—	TUN<5:0>						0000

图注： x = 复位时的未知值， — = 未实现 （读为 0）。复位值以十六进制显示。
注 1： RCON 寄存器复位值取决于复位类型。
2： OSCCON 寄存器复位值取决于 FOSC 配置位和复位类型。

表 3-31： NVM 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
NVMCON	0760	WR	WREN	WRERR	—	—	—	—	—	—	ERASE	—	—	NVMOP<3:0>				0000(1)
NVMKEY	0766	—	—	—	—	—	—	—	—	NVMKEY<7:0>								0000

图注： x = 复位时的未知值， — = 未实现 （读为 0）。复位值以十六进制显示。
注 1： 所示复位值仅适用于 POR。其他复位状态下的值取决于复位时存储器写操作或擦除操作的状态。

表 3-32： PMD 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	QEIMD	PWMMD	DCIMD	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	C2MD	C1MD	AD1MD	0000
PMD2	0772	IC8MD	IC7MD	IC6MD	IC5MD	IC4MD	IC3MD	IC2MD	IC1MD	OC8MD	OC7MD	OC6MD	OC5MD	OC4MD	OC3MD	OC2MD	OC1MD	0000
PMD3	0774	T9MD	T8MD	T7MD	T6MD	—	—	—	—	—	—	—	—	—	—	I2C2MD	AD2MD	0000

图注： x = 复位时的未知值， — = 未实现 （读为 0）。复位值以十六进制显示。

© 2007 Microchip Technology Inc.

3.2.7 软件堆栈

除了用作工作寄存器外，dsPIC33FJXXXGPX06/X08/X10 器件中的 W15 寄存器也可用作软件堆栈指针。堆栈指针总是指向第一个可用的空字，并且从低地址向高地址方向增长。它在弹出堆栈之前递减，而在压入堆栈之后递增，如图 3-6 所示。对于任何 CALL 指令时的 PC 压栈，在压入堆栈之前，PC的MSb要进行零扩展，从而确保MSb始终是清零的。

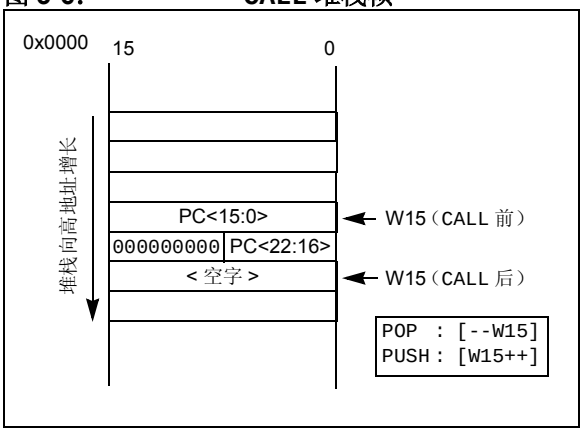
注：	在异常处理期间，在将 PC 压入堆栈之前，要先将 PC 的 MSb 与 SRL 寄存器组合在一起。

堆栈指针限制寄存器（SPLIM）与堆栈指针相关联，它设置堆栈上边界的值。复位时 SPLIM 不被初始化。与堆栈指针的情况一样， SPLIM<0> 被强制为 0，因为所有的堆栈操作必须是字对齐的。每当使用 W15 作为源指针或目标指针产生 EA 时，有效地址会与 SPLIM 中的值进行比较。如果堆栈指针 （W15）的内容与 SPLIM 寄存器的内容相等，则会执行压栈操作而不产生堆栈错误陷阱，但在随后的压栈操作时将会产生堆栈错误陷阱。这样的话，当堆栈增长超过 RAM 中地址 0x2000 时，如果要想产生堆栈错误陷阱，用值 0x1FFE 来初始化 SPLIM 即可。

类似地，当堆栈指针地址小于 0x0800 时，就会产生堆栈指针下溢（堆栈错误）陷阱。这可防止堆栈进入特殊功能寄存器 （SFR）空间。

在对 SPLIM 寄存器进行写操作之后，不应紧跟着使用 W15 进行间接读操作的指令。

图 3-6： CALL 堆栈帧



3.2.8 数据 RAM 保护功能

dsPIC33F 系列产品支持数据 RAM 保护功能，允许使用引导和安全代码段安全性保护 RAM 段。BS 的安全 RAM 段（Secure RAM Segment for BS，BSRAM）使能时，只能从引导段闪存代码进行访问。RAM 的安全 RAM 段（Secure RAM Segment for RAM， SSRAM）使能时，只能从安全段闪存代码进行访问。表 3-1 对 BSRAM 和 SSRAM SFR 进行了概括。

3.3 指令寻址模式

表 3-33 给出了基本的寻址模式，这些寻址模式经过优化可以支持各指令的具体功能。MAC 类指令中提供的寻址模式与其他指令类型中的寻址模式略有不同。

3.3.1 文件寄存器指令

大多数文件寄存器指令使用一个 13 位地址字段（f）来直接寻址数据存储器中的前 8192 字节 （Near 数据空间）。大多数文件寄存器指令使用工作寄存器 W0， W0 在这些指令中表示为 WREG。目标寄存器通常是同一个文件寄存器或 WREG （MUL 指令除外，它把结果写入寄存器或寄存器对）。使用 MOV 指令能够获得更大的灵活性，可以访问整个数据空间。

3.3.2 MCU 指令

三操作数 MCU 指令的形式是：

操作数 3 = 操作数 1< 功能 > 操作数 2

其中，操作数 1 始终是工作寄存器（即，寻址模式只能是寄存器直接寻址），称为 Wb。操作数 2 可以是一个 W 寄存器，取自数据存储器或一个 5 位立即数。结果可以被保存在 W 寄存器或数据存储单元中。MCU 指令支持以下寻址模式：

- 寄存器直接寻址
- 寄存器间接寻址
- 执行后修改的寄存器间接寻址
- 执行前修改的寄存器间接寻址
- 5 位或 10 位立即数寻址

注：	并非所有指令都支持上述所有的寻址模式。各条指令可能支持这些寻址模式中的某些模式。

表 3-33： 支持的基本寻址模式

寻址模式	说明
文件寄存器直接寻址	明确指定文件寄存器的地址。
寄存器直接寻址	直接访问寄存器的内容。
寄存器间接寻址	Wn 的内容形成 EA。
执行后修改的寄存器间接寻址	Wn 的内容形成 EA。然后用一个常量值来修改 Wn （递增或递减）。
执行前修改的寄存器间接寻址	先用一个有符号常量值修改 Wn （递增或递减），再由此时的 Wn 内容形成 EA。
带寄存器偏移量的寄存器间接寻址	Wn 和 Wb 的和形成 EA。
带立即数偏移量的寄存器间接寻址	Wn 和立即数的和形成 EA。

3.3.3 传送指令和累加器指令

与其他指令相比，传送指令和 DSP 累加器类指令提供了更为灵活的寻址模式。除了大多数 MCU 指令支持的寻址模式以外，传送和累加器指令还支持带寄存器偏移量的寄存器间接寻址模式，这也叫做寄存器变址寻址模式。

注： 对于 MOV 指令，指令中指定的寻址模式对于源寄存器和目标寄存器 EA，可以是不同的。然而，4 位 Wb（寄存器偏移量）字段为源寄存器和目标寄存器所共用 （但通常只由其中之一使用）。

概括地说，传送和累加器指令支持以下寻址模式：

- 寄存器直接寻址
- 寄存器间接寻址
- 执行后修改的寄存器间接寻址
- 执行前修改的寄存器间接寻址
- 带寄存器偏移量的寄存器间接寻址 （变址寻址）
- 带立即数偏移量的寄存器间接寻址
- 8 位立即数寻址
- 16 位立即数寻址

注： 并非所有指令都支持上述所有的寻址模式。各条指令可能支持这些寻址模式中的某些模式。

3.3.4 MAC 指令

双源操作数 DSP 指令（CLR、ED、EDAC、MAC、MPY、MPY.N、MOVSAC 和 MSC），也称为 MAC 指令，它们使用一组简化的寻址模式，允许用户通过寄存器间接寻址表高效地对数据指针进行操作。

双源操作数预取寄存器必须是集合 {W8, W9, W10, W11} 的成员。对于数据读取操作，W8 和 W9 始终用于 X RAGU，而 W10 和 W11 始终用于 Y AGU。从而，产生的有效地址（无论是在修改之前还是之后），对于 W8 和 W9 必须是 X 数据空间中的有效地址，对于 W10 和 W11 则必须是 Y 数据空间中的有效地址。

注： 带寄存器偏移量的寄存器间接寻址模式仅可用于 W9 （在 X 空间中）和 W11 （在 Y 空间中）。

概括地说， MAC 类指令支持以下寻址模式：

- 寄存器间接寻址
- 执行后修改 （修改量为 2）的寄存器间接寻址
- 执行后修改 （修改量为 4）的寄存器间接寻址
- 执行后修改 （修改量为 6）的寄存器间接寻址
- 带寄存器偏移量的寄存器间接寻址 （变址寻址）

3.3.5 其他指令

除了上述的各种寻址模式之外，一些指令使用各种长度的立即数。例如， BRA （转移）指令使用 16 位有符号立即数常量来直接指定转移的目标，而 DISI 指令则使用一个 14 位无符号立即数字段。在一些指令中，例如 ADD Acc，操作数的来源和运算结果已经暗含在操作码中。某些操作，例如 NOP，没有任何操作数。

3.4 模寻址

模寻址模式是一种使用硬件来自动支持循环数据缓冲区的方法。目的是在执行紧密循环代码时（这在许多 DSP 算法中很典型），不需要用软件来执行数据地址边界检查。

可以在数据空间或程序空间中进行模寻址 （因为这两种空间的数据指针机制本质上是相同的）。每个 X （也提供指向程序空间的指针）和 Y 数据空间中都可支持一个循环缓冲区。模寻址可以对任何 W 寄存器指针进行操

© 2007 Microchip Technology Inc.

作。然而，最好不要将 W14 或 W15 用于模寻址，因为这两个寄存器分别用作堆栈帧指针和堆栈指针。

总的来说，任何特定的循环缓冲区只能配置为单向工作，因为根据缓冲区的方向，对缓冲区起始地址（对于递增缓冲区）或结束地址（对于递减缓冲区）有某些限制。

使用限制的唯一例外是那些长度为 2 的幂的缓冲区。这些缓冲区满足起始和结束地址判据，它们可以双向工作（即，在低地址边界和高地址边界上都将进行地址边界检查）。

3.4.1 起始地址和结束地址

模寻址机制要求指定起始和结束地址，并将它们装入 16 位模缓冲区地址寄存器中：XMODSRT、 XMODEND、YMODSRT 和 YMODEND （见表 3-1）。

注：	Y 空间模寻址的 EA 计算使用字长度的数据（每个 EA 的 LSb 始终清零）。

循环缓冲区的长度没有直接指定。它由相应的起始和结束地址之差决定。循环缓冲区的最大长度为 32K 字（64 KB）。

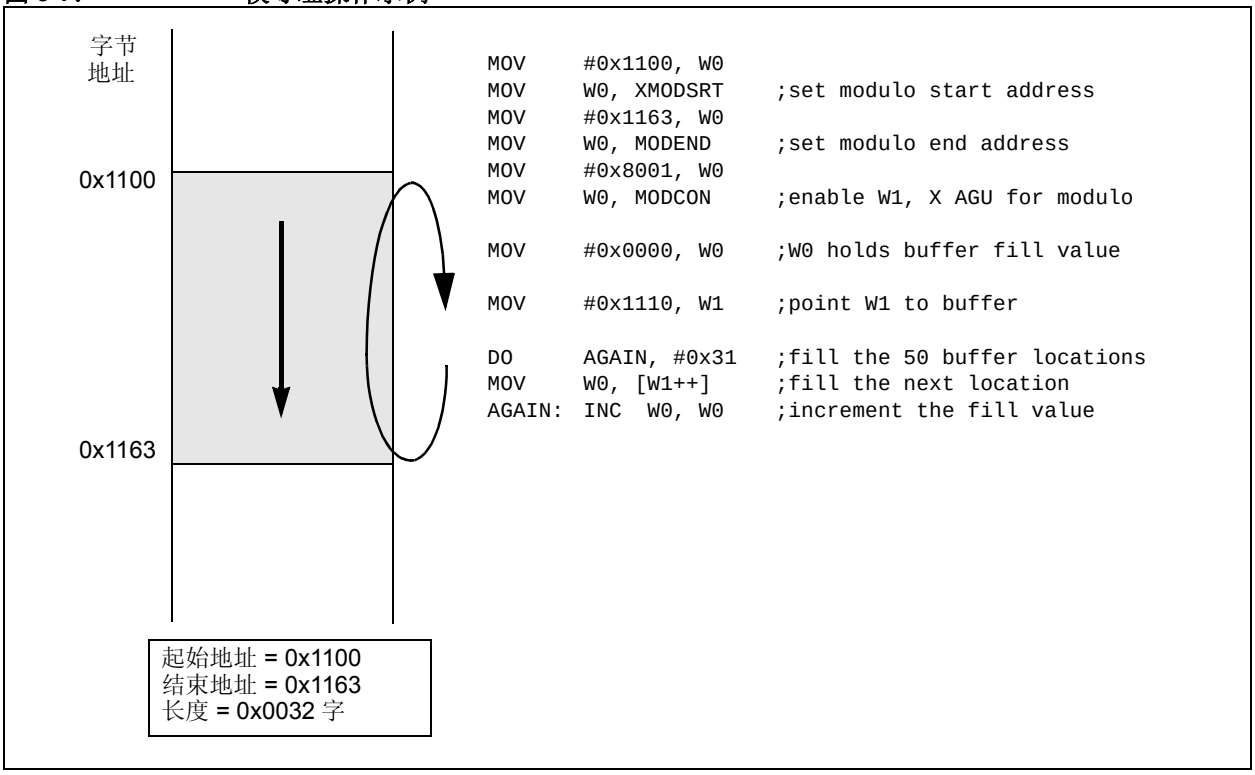
3.4.2 W 地址寄存器选择

模寻址和位反转寻址控制寄存器 MODCON<15:0> 包含使能标志以及指定 W 地址寄存器的 W 寄存器字段。XWM 和 YWM 字段选择对哪些寄存器进行模寻址。如果 XWM = 15，则禁止 X RAGU 和 X WAGU 模寻址。类似地，如果 YWM = 15，则禁止 Y AGU 模寻址。

要进行模寻址的 X 地址空间指针 W 寄存器（XWM）存储在 MODCON<3:0> 中（见表 3-1）。当 XWM 被设置为除15之外的任何值且XMODEN位（MODCON<15>）置 1 时， X 数据空间的模寻址被使能。

要对其进行模寻址的Y地址空间指针W寄存器（YWM）存储在 MODCON<7:4> 中。当 YWM 被设置为除 15 之外的任何值且 YMODEN 位（MODCON<14>）置 1 时，Y 数据空间的模寻址被使能。

图 3-7： 模寻址操作示例



3.4.3 模寻址的应用

模寻址可以应用于任何与 W 寄存器相关的有效地址（EA）计算中。重要的是要意识到，地址边界检查功能不仅会检查地址是否正好在地址边界上，而且会检查地址是否大于上边界地址 （对于递增缓冲区）、或小于下边界地址 （对于递减缓冲区）。因此，地址变化可能会越过边界，但仍然可以正确调整。

注：	只有在使用执行前修改或执行后修改寻址模式来计算有效地址时，模修正有效地址才被写回寄存器。如果使用了地址偏移量（例如，[W7+W2]），会进行模地址修正，但寄存器的内容保持不变。

3.5 位反转寻址

位反转寻址模式用来简化基 2 FFT 算法的数据重新排序。它为 X AGU 所支持，仅限于数据写入。

地址修改量可以是常数或寄存器的内容，可视为将其位顺序反转。源地址和目标地址仍然是正常的顺序。于是，唯一需要反转的操作数就是地址修改量。

3.5.1 位反转寻址的实现

当满足下列条件时，使能位反转寻址模式：

1. MODCON 寄存器中 BWM 位 （W 寄存器选择）的值是除 15 以外的任何值 （不能使用位反转寻址访问堆栈）。
2. XBREV 寄存器中的 BREN 位置 1。
3. 使用的寻址模式是预递增或后递增的寄存器间接寻址模式。

如果位反转缓冲区的长度为 $M = 2^N$ 字节，则数据缓冲区起始地址的最后 “N”位必须为零。

XB<14:0> 是位反转地址修改量或 “中心点”（pivot point），通常是一个常数。对于 FFT 计算，其值等于 FFT 数据缓冲区长度的一半。

注：	所有位反转 EA 的计算都使用字数据 （每个 EA 的 LSb 始终清零）。为了产生兼容（字节）地址，要相应地调整 XB 的值。

使能位反转寻址时，仅对预递增或后递增的寄存器间接寻址、且仅对字长度数据写入，才会进行位反转寻址。对于任何其他寻址模式或对于字节长度数据，不会进行位反转寻址，而是生成正常的地址。在进行位反转寻址时，W 地址指针的增量将始终加上地址修改量（XB），与寄存器间接寻址模式相关的偏移量将被忽略。此外，由于要求是字数据，EA 的 LSb 被忽略（且始终清零）。

注：	不应同时使能模寻址和位反转寻址。如果用户试图这么做的话，对于 X WAGU，位反转寻址将优先， X WAGU 模寻址将被禁止。然而，在 X RAGU 中，模寻址继续起作用。

如果通过将 BREN 位（XBREV<15>）置 1 使能了位反转寻址，那么，在写 XBREV 寄存器之后，不应立即进行要使用被指定为位反转指针的 W 寄存器的间接读操作。

© 2007 Microchip Technology Inc.

图 3-8：　　位反转地址示例

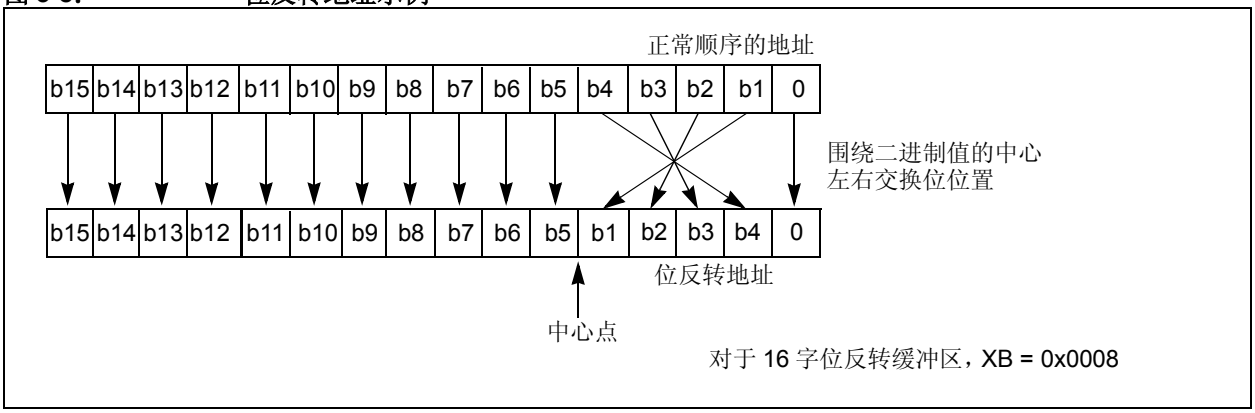

正常顺序的地址
b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 0
围绕二进制值的中心
左右交换位位置
b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b1 b2 b3 b4 0
位反转地址
中心点
对于 16 字位反转缓冲区，XB = 0x0008

表 3-34：　　位反转地址序列 （16 项）

正常地址					位反转地址				
A3	A2	A1	A0	十进制	A3	A2	A1	A0	十进制
0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	8
0	0	1	0	2	0	1	0	0	4
0	0	1	1	3	1	1	0	0	12
0	1	0	0	4	0	0	1	0	2
0	1	0	1	5	1	0	1	0	10
0	1	1	0	6	0	1	1	0	6
0	1	1	1	7	1	1	1	0	14
1	0	0	0	8	0	0	0	1	1
1	0	0	1	9	1	0	0	1	9
1	0	1	0	10	0	1	0	1	5
1	0	1	1	11	1	1	0	1	13
1	1	0	0	12	0	0	1	1	3
1	1	0	1	13	1	0	1	1	11
1	1	1	0	14	0	1	1	1	7
1	1	1	1	15	1	1	1	1	15

3.6 程序存储空间与数据存储空间的接口

dsPIC33FJXXXGPX06/X08/X10 架构采用 24 位宽的程序空间和 16 位宽的数据空间。该架构也是一种改进型哈佛结构，这意味着数据也能存放在程序空间内。要成功使用该数据，在访问数据时必须确保这两种存储空间中的信息是对齐的。

除了正常执行外， dsPIC33FJXXXGPX06/X08/X10 架构还提供了两种可在操作过程中访问程序空间的方法：

- 使用表指令访问程序空间中任意位置的各个字节或字
- 将程序空间的一部分重新映射到数据空间 （程序空间可视性）

表指令允许应用程序读写程序存储器的一小块区域。这一功能对于访问需要随时更新的数据表来说非常理想。也可通过表操作访问一个程序字的所有字节。重映射方法允许应用程序访问一大块数据，但只限于读操作，它非常适合于在一个大的静态数据表中进行查找。这一方法只能访问程序字的低位字。

3.6.1 对程序空间进行寻址

由于数据和程序空间的地址范围分别为 16 位和 24 位，因此需要一个从 16 位数据寄存器创建一个 23 位或 24 位程序地址的方法。方法取决于所采用的接口方式。

对于表操作，使用 8 位的表页寄存器（TBLPAG）定义程序空间内一个 32K 字的区域。这与 16 位 EA 组合形成了一个完整的 24 位程序空间地址。在这种地址形式下，TBLPAG 的最高位用来决定操作是发生在用户存储区中（TBLPAG<7> = 0）还是配置存储区中（TBLPAG<7> = 1）。

对于重映射操作，使用 8 位的程序空间可视性页寄存器（PSVPAG）定义程序空间中的 16K 字页。当 EA 的最高位为 1 时， PSVPAG 与 EA 的低 15 位组合形成一个 23 位的程序空间地址。与表操作不同，重映射操作被严格限制在用户存储区中。

表 3-35 和图 3-9 显示了如何通过表操作和重映射访问来从数据 EA 创建程序 EA。这里，P<23:0> 指的是程序空间字；而 D<15:0> 指的是数据空间字。

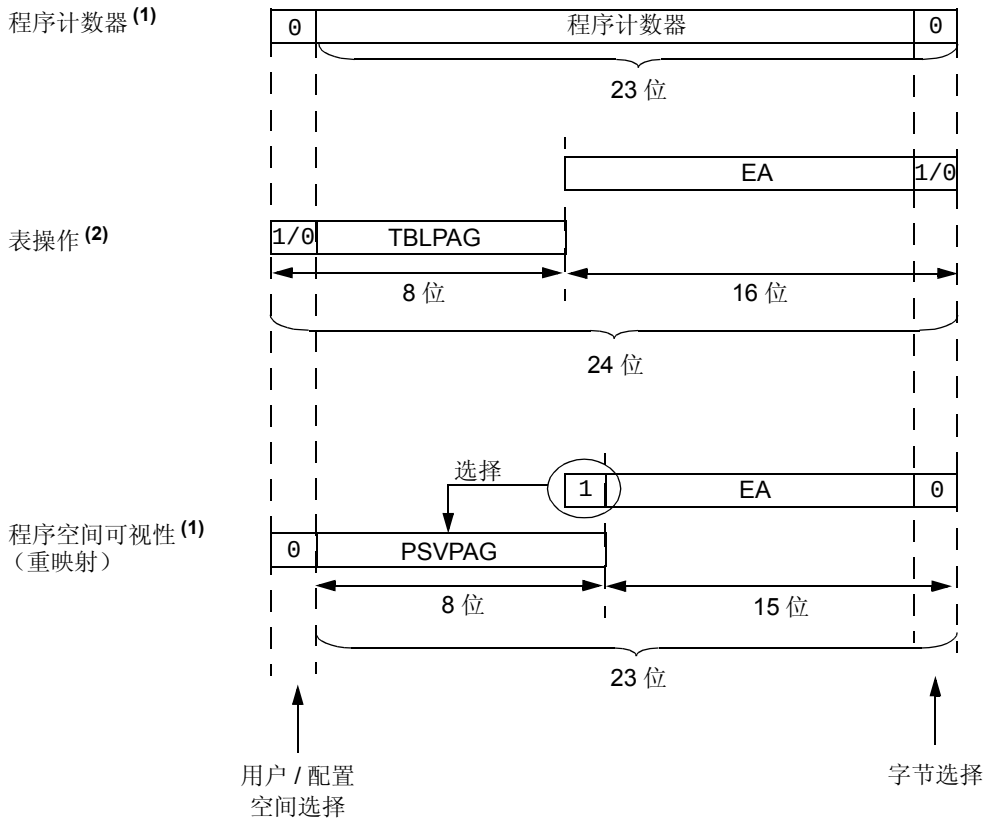
表 3-35： 程序空间地址构成

访问类型	访问空间	程序空间地址				
		<23>	<22:16>	<15>	<14:1>	<0>
指令访问（代码执行）	用户	0	PC<22:1>			0
		0xx xxxx xxxx xxxx xxxx xxx0				
TBLRD/TBLWT（读 / 写字节或字）	用户	TBLPAG<7:0>		数据 EA<15:0>		
		0xxx xxxx		xxxx xxxx xxxx xxxx		
	配置	TBLPAG<7:0>		数据 EA<15:0>		
		1xxx xxxx		xxxx xxxx xxxx xxxx		
程序空间可视性（块重映射 / 读）	用户	0	PSVPAG<7:0>		数据 EA<14:0>(1)	
		0	xxxx xxxx		xxx xxxx xxxx xxxx	

注 1： 在这种情况下，数据 EA<15> 始终为 1，但并不用它来计算程序空间地址。地址的 bit 15 为 PSVPAG<0>。

© 2007 Microchip Technology Inc.

图 3-9：　　访问程序空间内数据的地址生成方式



注　1： 程序空间地址的最低位（LSb）始终为 0，从而确保程序和数据空间中的数据是字对齐的。

2： 不要求表操作是字对齐的。允许对配置存储空间执行表读操作。

3.6.2 使用表指令访问程序存储器中的数据

TBLRDL 和 TBLWTL 指令提供了读或写程序空间内任何地址的低位字的直接方法，无需通过数据空间。TBLRDH和TBLWTH指令是可以把一个程序空间字的高8位作为数据读写的唯一方法。

对于每个连续的 24 位程序字，PC 的递增量为 2。这使得程序存储器地址能够直接映射到数据空间地址。于是，程序存储器可以看作是两个 16 位字宽的地址空间，它们并排放置，具有相同的地址范围。TBLRDL 和 TBLWTL 访问存有数据低位字的空间，而 TBLRDH 和 TBLWTH 则访问存有最高数据字节的空间。

提供了两条表指令来对程序空间执行字节或字（16 位）大小的数据读写。读和写都可以采用字节或字操作的形式。

1. TBLRDL（表读低位字）：在字模式下，该指令将程序空间地址的低位字（P<15:0>）映射到数据地址（D<15:0>）中。

   在字节模式下，低位程序字的高字节或低字节被映射到数据地址的低字节中。当字节选择位为 1 时映射高字节；当字节选择位为 0 时映射低字节。

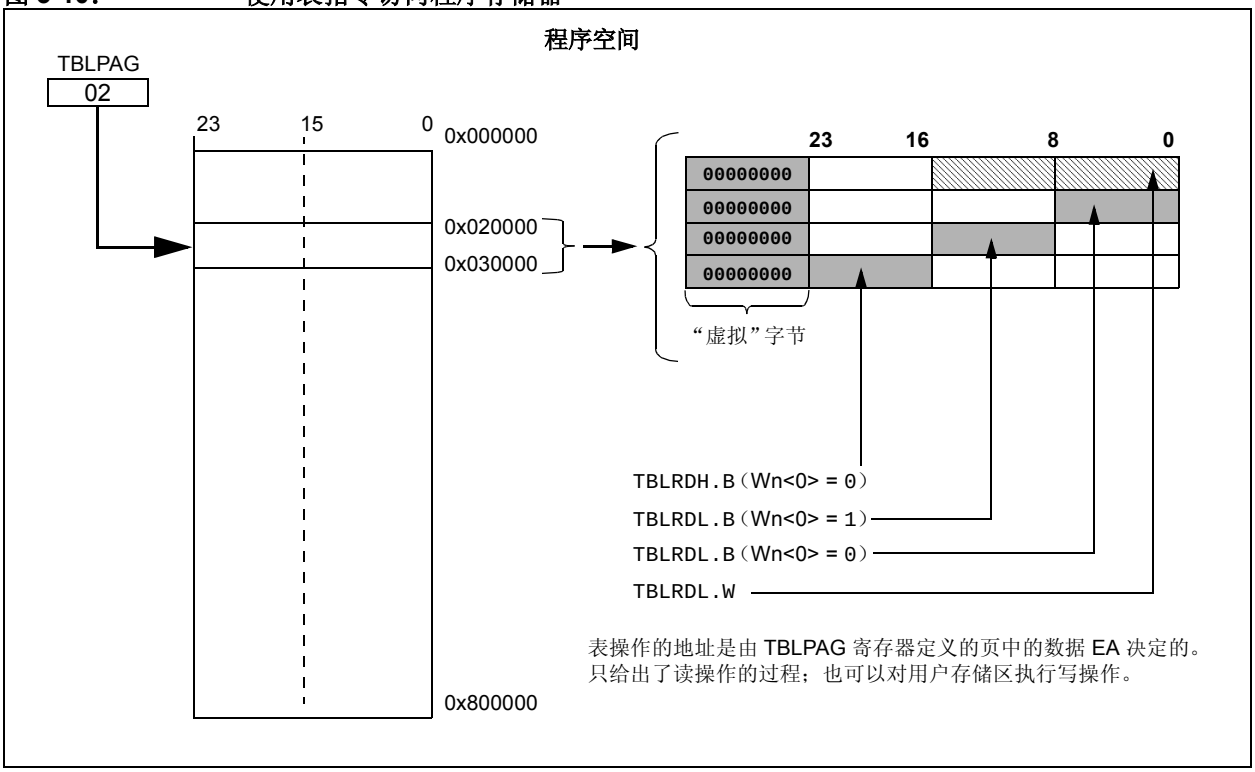
2. TBLRDH（表读高位字）：在字模式下，该指令将程序地址的整个高位字（P<23:16>）映射到数据地址中。注意，D<15:8> 为“虚拟”字节，它始终为 0。

   在字节模式下，该指令将程序字的高字节或低字节映射到数据地址的D<7:0>中，就如同TBLRDL指令。注意，当选择高位“虚拟”字节（字节选择位 = 1）时，数据将始终为 0。

表指令TBLWTH和TBLWTL以类似的方式向程序空间地址写入各字节或字。**第 4.0 节“闪存程序存储器”**对这两条指令的详细操作给出了说明。

对于所有的表操作，要访问程序存储空间的哪个区域是由表页寄存器（TBLPAG）决定的。TBLPAG 可寻址器件的整个程序存储空间，包括用户和配置空间。当 TBLPAG<7> = 0 时，表页位于用户存储空间中。当 TBLPAG<7> = 1 时，表页位于配置存储空间中。

图 3-10： 使用表指令访问程序存储器



© 2007 Microchip Technology Inc.

3.6.3 使用程序空间可视性读程序存储器中的数据

可选择将数据空间的高32 KB映射到程序空间中的任何16K字页中。这提供了通过数据空间对存储的常量数据的透明访问，而无需使用特殊指令（即TBLRDL/H）。

如果数据空间EA的最高有效位为1，并且程序空间可视性使能（方法是将内核控制寄存器中的PSV位（CORCON<2>）置1）时，就能通过数据空间访问程序空间。由程序空间可视性页寄存器（PSVPAG）决定要被映射到数据空间中的程序存储空间的位置。这一8位的寄存器定义程序空间中256个可能的16K字页中的一个。事实上，PSVPAG作为程序存储地址的高8位，而EA的15位则作为地址的低位。请注意，对于每个程序存储字，PC都将递增2，数据空间地址的低15位将直接映射到相应程序空间地址的低15位。

正在执行的指令，如果要对这个区域进行数据读取的话，就需要增加一个额外的指令周期，因为需要进行两次程序存储空间读取操作。

尽管大于或等于8000h的每个数据空间地址直接映射到对应的程序存储器地址（见图3-11），但只使用24位程序字的低16位来存放数据。所有用来存放数据的程序存储单元的高8位都应被设置为1111 1111或0000 0000，强制为一条NOP指令，从而避免了可能出现意外执行这一区域内代码的情况。

注： 在表读 / 写期间，暂时禁止PSV访问。

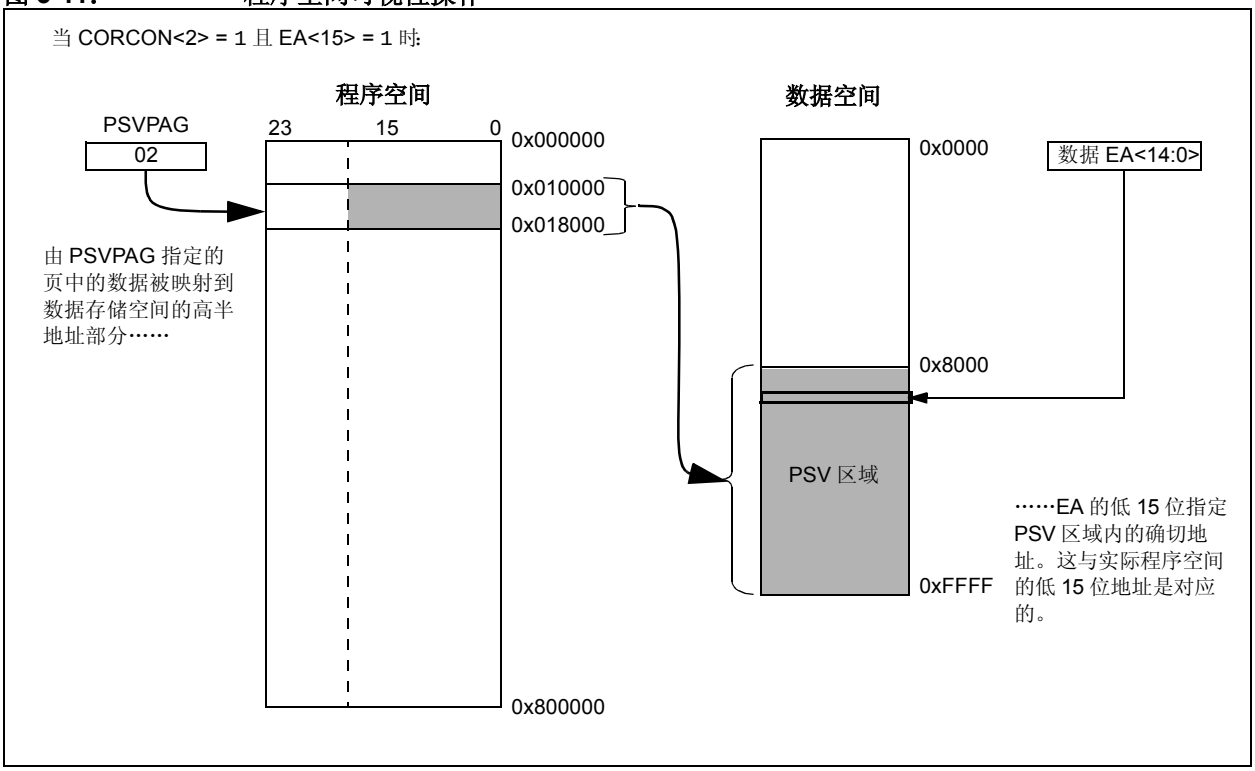
对于使用PSV而又在REPEAT循环外执行的操作，MOV和MOV.D指令除了规定的执行时间之外，还需要一个额外的指令周期。其他所有指令都需要在规定的执行时间之外额外增加两个指令周期。

对于使用PSV而又在REPEAT循环内执行的操作，下列情况，除了规定的指令执行时间之外，还需要两个额外的指令周期：

- 在第一次迭代中执行的指令
- 在最后一次迭代中执行的指令
- 由于中断而退出循环之前执行的指令
- 中断得到处理后再次进入循环时执行的指令

REPEAT循环的所有其他各次迭代，都允许使用PSV访问数据的指令在一个周期内执行。

图 3-11： 程序空间可视性操作



注：

© 2007 Microchip Technology Inc.

4.0 闪存程序存储器

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10 器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

dsPIC33FJXXXGPX06/X08/X10 器件包含用于存储和执行应用代码的内部闪存程序存储器。在整个 VDD 范围内，正常操作期间，存储器都是可读写可擦除的。

可采用两种方式对闪存存储器进行编程：

1. 在线串行编程（In-Circuit Serial Programming™， ICSP™）功能
2. 运行时自编程（Run-Time Self-Programming，RTSP）

ICSP 允许在最终的应用电路中对 dsPIC33FJXXXGPX06/X08/X10 器件进行串行编程。只需要使用 5 根线就可以完成编程，它们分别是编程时钟线、编程数据线（以下备用编程引脚对之一：PGC1/PGD1、PGC2/PGD2 或 PGC3/PGD3）、电源线（VDD）、接地线（VSS）和主复位线（MCLR）。这允许用户在生产电路板时使用未编程器件，仅在产品交付之前才对数字信号控制器进行编程，从而可以使用最新版本的固件或者定制固件进行编程。

使用 TBLRD（表读）和 TBLWT（表写）指令来实现 RTSP。使用 RTSP，用户可以一次将 64 条指令（192 字节）的块（或“行”）或单个程序存储字写入程序存储器，也可以一次擦除 512 条指令（1536 字节）的块（或“页”）。

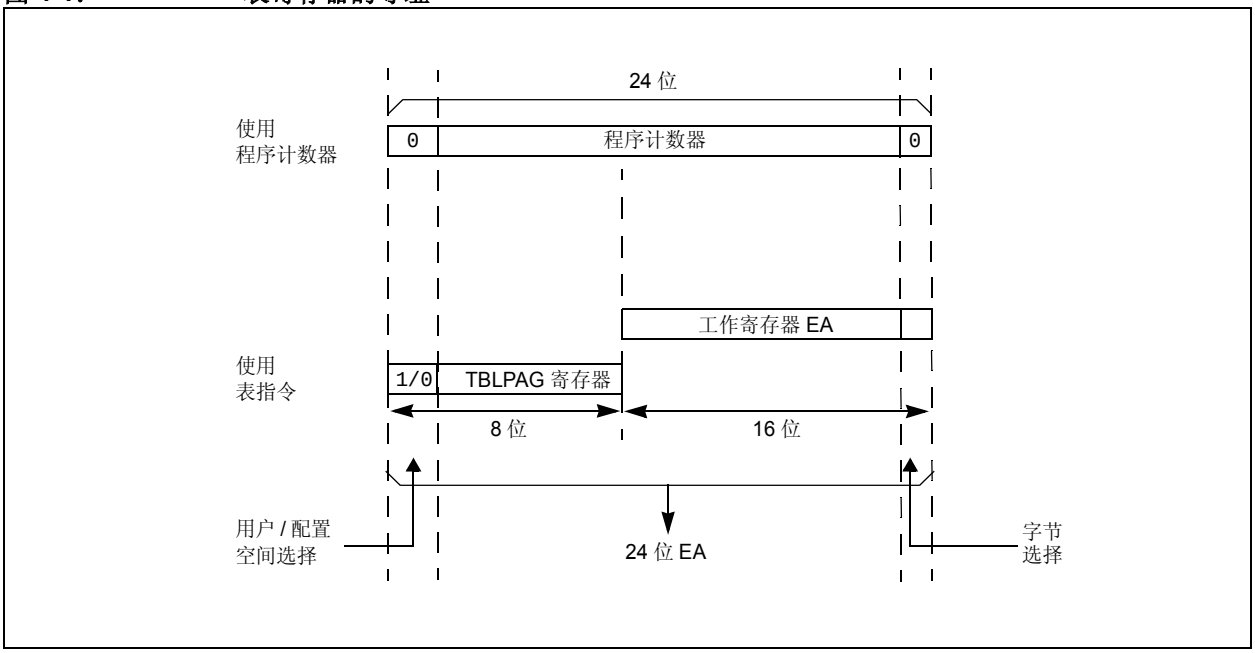
4.1 表指令和闪存编程

闪存存储器的编程都是用表读和表写指令实现的，与使用的编程方法无关。这些指令允许器件在正常工作模式下通过数据存储器直接读写程序存储空间。程序存储器中 24 位目标地址由 TBLPAG 寄存器的 bit<7:0> 和表指令中指定 W 寄存器中的有效地址（EA）组成，如图 4-1 所示。

TBLRDL 和 TBLWTL 指令用来读或写程序存储器的 bit<15:0>。TBLRDL 和 TBLWTL 能以字或字节模式访问程序存储器。

TBLRDH 和 TBLWTH 指令用来读或写程序存储器的 bit<23:16>。TBLRDH 和 TBLWTH 同样能以字或字节模式访问程序存储器。

图 4-1： 表寄存器的寻址



4.2 RTSP 工作原理

dsPIC33FJXXXGPX06/X08/X10 闪存程序存储器阵列是由 64 条指令或 192 字节的行组成的。 RTSP 允许用户一次擦除由 8 行数据（512 条指令）组成的程序存储器页，一次编程一行或一个字。**表 24-12，直流特性：程序存储器**给出了典型的擦除和编程次数。 8 行擦除页和单行写入行都是边界对齐的，从程序存储器起始地址开始，分别到 1536 字节边界和 192 字节边界。

程序存储器实现了保持缓冲器，它能缓冲 64 条指令的编程数据。在实际编程操作前，必须将待写数据顺序装入缓冲器。装入的指令字必须始终来自一组 64 个指令字的边界。

RTSP 编程的基本步骤是先建立一个表指针，然后执行一系列 TBLWT 指令将数据装入缓冲器。编程是通过将 NVMCON 寄存器中的控制位置 1 执行的。指令的装入总共需要 64 条 TBLWTL 和 TBLWTH 指令。

所有的表写操作都是单字写操作 （2 个指令周期），因为只写缓冲器。编程每一行需要一个编程周期。

4.3 控制寄存器

有两个 SFR 用于读写程序闪存存储器：

- **NVMCON：闪存存储器控制寄存器**
- **NVMKEY：非易失性存储器密钥寄存器**

NVMCON 寄存器 （寄存器 4-1）控制要擦除的块、要编程的存储器类型，以及编程周期的开始。

NVMKEY （寄存器 4-2）是一个只写寄存器，用于写保护。要启动编程或擦除序列，用户必须连续地将 55h 和 AAh 写入 NVMKEY 寄存器。更多详细信息，请参见**第 4.4 节 “编程操作”**。

4.4 编程操作

在 RTSP 模式下，对内部闪存进行编程或擦除需要执行完整的编程过程。编程操作的持续时间通常为 4 ms，在此期间处理器暂停等待操作完成。将 WR 位（NVMCON<15>）置 1 启动操作，当操作完成时 WR 位会自动清零。

© 2007 Microchip Technology Inc.

寄存器 4-1：　　NVMCON：闪存存储器控制寄存器

R/SO-0(1)	R/W-0(1)	R/W-0(1)	U-0	U-0	U-0	U-0	U-0
WR	WREN	WRERR	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0(1)	U-0	U-0	R/W-0(1)	R/W-0(1)	R/W-0(1)	R/W-0(1)
—	ERASE	—	—	NVMOP<3:0>(2)			
bit 7							bit 0

图注：	SO = 只可置 1 位		
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**WR：**写控制位
1 = 启动闪存存储器编程或擦除操作。该操作是自定时的，一旦操作完成该位即由硬件清零。
0 = 编程或擦除操作完成，并处于停止状态

bit 14　**WREN：**写使能位
1 = 使能闪存编程 / 擦除操作
0 = 禁止闪存编程 / 擦除操作

bit 13　**WRERR：**写序列错误标志位
1 = 试图执行不合法的编程或擦除序列，或者发生终止（试图将 WR 位置 1 时自动置 1 该位）
0 = 编程或擦除操作正常完成

bit 12-7　**未实现：**读为 0

bit 6　**ERASE：**擦除 / 编程使能位
1 = 在下一条 WR 命令时执行 NVMOP<3:0> 指定的擦除操作
0 = 在下一条 WR 命令时执行 NVMOP<3:0> 指定的编程操作

bit 5-4　**未实现：**读为 0

bit 3-0　**NVMOP<3:0>：**NVM 操作选择位 (2)

如果 ERASE = 1：
1111 = 存储器批量擦除操作
1110 = 保留
1101 = 擦除通用段
1100 = 擦除安全段
1011 = 保留
0011 = 无操作
0010 = 存储器页擦除操作
0001 = 无操作
0000 = 擦除单个配置寄存器字节

如果 ERASE = 0：
1111 = 无操作
1110 = 保留
1101 = 无操作
1100 = 无操作
1011 = 保留
0011 = 存储器字编程操作
0010 = 无操作
0001 = 存储器行编程操作
0000 = 编程单个配置寄存器字节

注　1： 这些位只能在 POR 时复位。
2： NVMOP<3:0> 的所有其他组合均未实现。

寄存器 4-2：　　NVMKEY：非易失性存储器密钥寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

W-0	W-0	W-0	W-0	W-0	W-0	W-0	W-0
NVMKEY<7:0>							
bit 7							bit 0

图注：	SO = 只可置 1 位		
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-8　**未实现：**读为 0

bit 7-0　**NVMKEY<7:0>：**密钥寄存器 （只写）位

© 2007 Microchip Technology Inc.

4.4.1 闪存程序存储器的编程算法

用户可以一次对闪存存储器的一行进行编程。要实现该操作，首先需要擦除包含该行在内的一个 8 行大小的页。一般过程如下：

1. 读取程序存储器的 8 行 （512 条指令），并存储在数据 RAM 中。
2. 用所需的新数据更新 RAM 中的程序数据。
3. 擦除程序块 （见例 4-1）：
   a) 将 NVMOP 位 （NVMCON<3:0>）设置为 0010，配置为块擦除操作。将 ERASE （NVMCON<6>）和 WREN （NVMCON<14>）位置 1。
   b) 将要被擦除的页的起始地址写入TBLPAG和W 寄存器。
   c) 将 55h 写入 NVMKEY。
   d) 将 AAh 写入 NVMKEY。
   e) 将 WR 位 （NVMCON<15>）置 1。擦除周期开始，在擦除周期中 CPU 会暂停。当擦除完成时， WR 位会自动清零。
4. 将数据 RAM 中的前 64 条指令写入程序存储器缓冲器 （见例 4-2）。
5. 将程序块写入闪存存储器：
   a) 将 NVMOP 位设置为 0001，配置为行编程操作。将 ERASE 位清零，将 WREN 位置 1。
   b) 将 55h 写入 NVMKEY。
   c) 将 AAh 写入 NVMKEY。
   d) 将 WR 位置 1。编程周期开始，在写周期中 CPU 会暂停。当闪存存储器写操作完成时，WR 位会自动清零。
6. 将 TBLPAG 中的值递增 1，使用数据 RAM 中下一个 64 条指令块重复步骤 4 和 5，直到所有 512 条指令被写回闪存存储器。

为防止意外操作，必须向 NVMKEY 写入启动序列从而允许执行擦除或编程操作。在执行编程命令后，用户必须等待一段编程时间，直至编程完成。紧跟编程启动序列后面的两条指令应为 NOP，如例 4-3 所示。

例 4-1： 擦除程序存储器页

```
; Set up NVMCON for block erase operation
        MOV     #0x4042, W0                 ;
        MOV     W0, NVMCON                  ; Initialize NVMCON
; Init pointer to row to be ERASED
        MOV     #tblpage(PROG_ADDR), W0     ;
        MOV     W0, TBLPAG                  ; Initialize PM Page Boundary SFR
        MOV     #tbloffset(PROG_ADDR), W0   ; Initialize in-page EA[15:0] pointer
        TBLWTL  W0, [W0]                    ; Set base address of erase block
        DISI    #5                          ; Block all interrupts with priority <7
                                            ; for next 5 instructions
        MOV     #0x55, W0
        MOV     W0, NVMKEY                  ; Write the 55 key
        MOV     #0xAA, W1                   ;
        MOV     W1, NVMKEY                  ; Write the AA key
        BSET    NVMCON, #WR                 ; Start the erase sequence
        NOP                                 ; Insert two NOPs after the erase
        NOP                                 ; command is asserted
```

例 4-2：　　装载写缓冲器

```
; Set up NVMCON for row programming operations
       MOV     #0x4001, W0                     ;
       MOV     W0, NVMCON                      ; Initialize NVMCON
; Set up a pointer to the first program memory location to be written
; program memory selected, and writes enabled
       MOV     #0x0000, W0                     ;
       MOV     W0, TBLPAG                      ; Initialize PM Page Boundary SFR
       MOV     #0x6000, W0                     ; An example program memory address
; Perform the TBLWT instructions to write the latches
; 0th_program_word
       MOV     #LOW_WORD_0, W2                 ;
       MOV     #HIGH_BYTE_0, W3                ;
       TBLWTL  W2, [W0]                        ; Write PM low word into program latch
       TBLWTH  W3, [W0++]                      ; Write PM high byte into program latch
; 1st_program_word
       MOV     #LOW_WORD_1, W2                 ;
       MOV     #HIGH_BYTE_1, W3                ;
       TBLWTL  W2, [W0]                        ; Write PM low word into program latch
       TBLWTH  W3, [W0++]                      ; Write PM high byte into program latch
;  2nd_program_word
       MOV     #LOW_WORD_2, W2                 ;
       MOV     #HIGH_BYTE_2, W3                ;
       TBLWTL  W2, [W0]                        ; Write PM low word into program latch
       TBLWTH  W3, [W0++]                      ; Write PM high byte into program latch
       •
       •
       •
; 63rd_program_word
       MOV     #LOW_WORD_31, W2                ;
       MOV     #HIGH_BYTE_31, W3               ;
       TBLWTL  W2, [W0]                        ; Write PM low word into program latch
       TBLWTH  W3, [W0++]                      ; Write PM high byte into program latch
```

例 4-3：　　启动编程序列

```
       DISI    #5                      ; Block all interrupts with priority <7
                                       ; for next 5 instructions
       MOV     #0x55, W0
       MOV     W0, NVMKEY              ; Write the 55 key
       MOV     #0xAA, W1               ;
       MOV     W1, NVMKEY              ; Write the AA key
       BSET    NVMCON, #WR             ; Start the erase sequence
       NOP                             ; Insert two NOPs after the
       NOP                             ; erase command is asserted
```


© 2007 Microchip Technology Inc.

5.0 复位

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

复位模块结合了所有复位源并控制器件的主复位信号SYSRST。下面列出了器件的复位源：

- POR：上电复位
- BOR：欠压复位
- MCLR：主复位引脚复位
- SWR：RESET 指令
- WDT：看门狗定时器复位
- TRAPR：陷阱冲突复位
- IOPUWR：非法操作码和未初始化的 W 寄存器复位

图 5-1 给出了复位模块的简化框图。

任何有效的复位源都会使 SYSRST 信号有效。很多与CPU 和外设相关的寄存器均会被强制为已知的复位状态。大多数寄存器都不受复位影响；它们的状态在POR时未知，而在所有其他复位时不变。

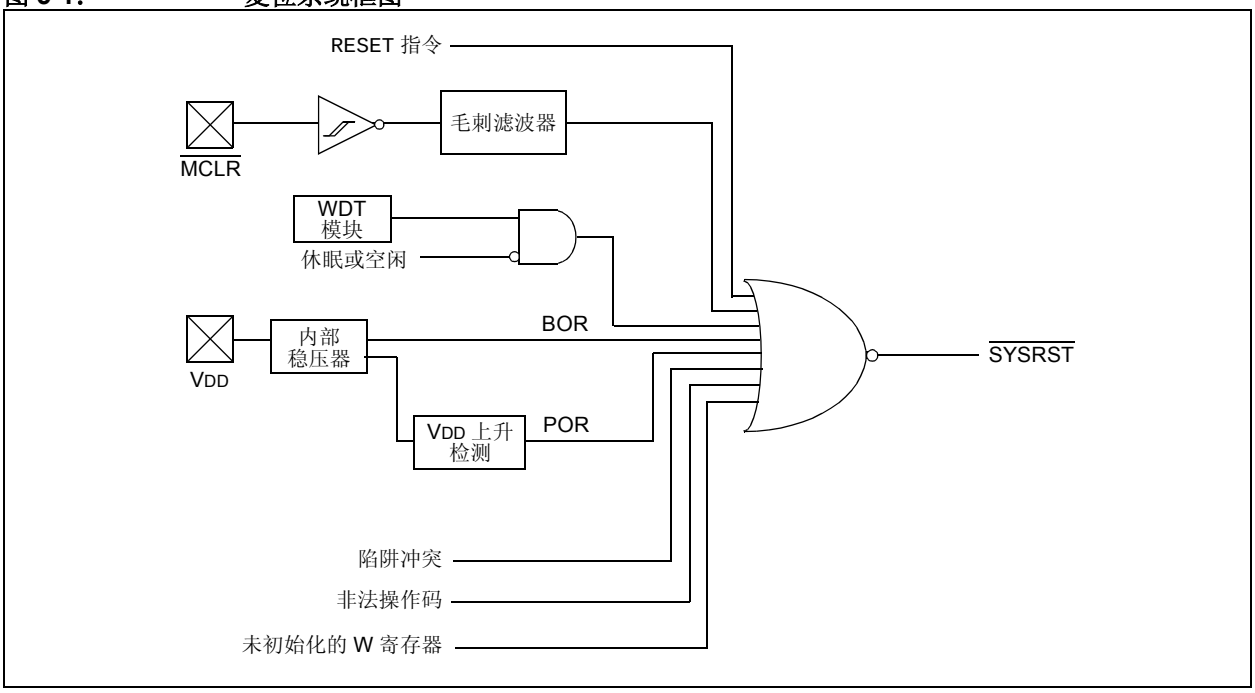
注： 如需了解寄存器复位状态的信息，请参见本手册中特定的外设或 CPU 章节。

任何类型的器件复位都会将RCON寄存器中相应的状态位置 1，以表明复位类型（见寄存器 5-1）。POR 将清零除 POR 位（RCON<0>）之外的所有位，POR 位在 POR 时置 1。用户可在代码执行过程中的任何时间置 1 或清零任何位。RCON 位仅用作状态位。在软件中将特定的复位状态位置 1 不会导致器件发生复位。

RCON寄存器还有与看门狗定时器和器件节能状态相关的其他位。本手册的其他章节中将讨论这些位的功能。

注： RCON 寄存器中的状态位应该在被读取后清零，这样在器件复位后的下一个 RCON 寄存器值才有意义。

图 5-1： 复位系统框图



寄存器 5-1：　　RCON：复位控制寄存器 (1)

R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	R/W-0
TRAPR	IOPUWR	—	—	—	—	—	VREGS
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1
EXTR	SWR	SWDTEN(2)	WDTO	SLEEP	IDLE	BOR	POR
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**TRAPR：**陷阱复位标志位
1 = 发生了陷阱冲突复位
0 = 未发生陷阱冲突复位

bit 14　**IOPUWR：**非法操作码或未初始化的 W 寄存器访问复位标志位
1 = 检测到非法操作码、非法地址模式或将未初始化的 W 寄存器用作地址指针而导致复位
0 = 未发生非法操作码或未初始化的 W 寄存器复位

bit 13-9　**未实现：**读为 0

bit 8　**VREGS：**休眠模式下稳压器待机位
1 = 在休眠模式下稳压器继续工作
0 = 在休眠模式下稳压器进入待机模式

bit 7　**EXTR：**外部复位 （$\overline{\text{MCLR}}$）引脚位
1 = 发生了主复位 （引脚）复位
0 = 未发生主复位 （引脚）复位

bit 6　**SWR：**软件复位 （指令）标志位
1 = 执行了 RESET 指令
0 = 未执行 RESET 指令

bit 5　**SWDTEN：**软件使能 / 禁止 WDT 位 (2)
1 = 使能 WDT
0 = 禁止 WDT

bit 4　**WDTO：**看门狗定时器超时标志位
1 = 发生了 WDT 超时
0 = 未发生 WDT 超时

bit 3　**SLEEP：**从休眠模式唤醒标志位
1 = 器件处于休眠模式
0 = 器件不处于休眠模式

bit 2　**IDLE：**从空闲模式唤醒标志位
1 = 器件处于空闲模式
0 = 器件不处于空闲模式

bit 1　**BOR：**欠压复位标志位
1 = 发生了欠压复位
0 = 未发生欠压复位

注　1：所有复位状态位都可以用软件置 1 或清零。用软件将这些位中的一位置 1 不会导致器件复位。
2：如果 FWDTEN 配置位为 1 （未编程），则 WDT 始终使能，而与 SWDTEN 位的设置无关。

© 2007 Microchip Technology Inc.

寄存器 5-1：　　RCON：复位控制寄存器 (1)（续）

bit 0　　**POR**：上电复位标志位
1 = 发生了上电复位
0 = 未发生上电复位

注　1： 所有复位状态位都可以用软件置 1 或清零。用软件将这些位中的一位置 1 不会导致器件复位。
2： 如果 FWDTEN 配置位为 1（未编程），则 WDT 始终使能，而与 SWDTEN 位的设置无关。

表 5-1：　　复位标志位操作

标志位	置 1 所表示的事件	清零所表示的事件
TRAPR （RCON<15>）	陷阱冲突事件	POR
IOPUWR （RCON<14>）	非法操作码或访问了未初始化的 W 寄存器	POR
EXTR （RCON<7>）	$\overline{\text{MCLR}}$ 复位	POR
SWR （RCON<6>）	RESET 指令	POR
WDTO （RCON<4>）	WDT 超时	PWRSAV 指令和 POR
SLEEP （RCON<3>）	PWRSAV #SLEEP 指令	POR
IDLE （RCON<2>）	PWRSAV #IDLE 指令	POR
BOR （RCON<1>）	BOR	—
POR （RCON<0>）	POR	—

注：　所有复位标志位均可由用户软件置 1 或清零。

5.1　复位时的时钟源选择

如果使能了时钟切换，器件复位时的系统时钟源选择如表 5-2 中所示。如果禁止了时钟切换功能，则总是根据振荡器配置位选择系统时钟源。更多详细信息，请参见**第 8.0 节 “振荡器配置”**。

表 5-2：　　不同复位类型的振荡器选择 （使能时钟切换功能）

复位类型	确定时钟源的方式
POR	振荡器配置位（FNOSC<2:0>）
BOR	
$\overline{\text{MCLR}}$	COSC 控制位（OSCCON<14:12>）
WDTR	
SWR	

5.2　器件复位时间

表 5-3 总结了各种类型器件复位的复位时间。系统复位信号 SYSRST 在 POR 和 PWRT 延时结束后发出。

器件实际开始执行代码的时间还取决于系统振荡器延时，它包括振荡器起振定时器 （OST）延时和 PLL 锁定时间。OST 和 PLL 锁定时间与相应的 $\overline{\text{SYSRST}}$ 延时同时发生。

FSCM 延时决定在 $\overline{\text{SYSRST}}$ 信号发出到 FSCM 开始监视系统时钟源的时间。

© 2007 Microchip Technology Inc.

表 5-3：　　各种器件复位的复位延时

复位类型	时钟源	$\overline{\text{SYSRST}}$ 延时	系统时钟延时	FSCM 延时	注
POR	EC, FRC, LPRC	TPOR + TSTARTUP + TRST	—	—	**1, 2, 3**
	ECPLL, FRCPLL	TPOR + TSTARTUP + TRST	TLOCK	TFSCM	**1, 2, 3, 5, 6**
	XT, HS, SOSC	TPOR + TSTARTUP + TRST	TOST	TFSCM	**1, 2, 3, 4, 6**
	XTPLL, HSPLL	TPOR + TSTARTUP + TRST	TOST + TLOCK	TFSCM	**1, 2, 3, 4, 5, 6**
BOR	EC, FRC, LPRC	TSTARTUP + TRST	—	—	**3**
	ECPLL, FRCPLL	TSTARTUP + TRST	TLOCK	TFSCM	**3, 5, 6**
	XT, HS, SOSC	TSTARTUP + TRST	TOST	TFSCM	**3, 4, 6**
	XTPLL, HSPLL	TSTARTUP + TRST	TOST + TLOCK	TFSCM	**3, 4, 5, 6**
$\overline{\text{MCLR}}$	任何时钟	TRST	—	—	**3**
WDT	任何时钟	TRST	—	—	**3**
软件	任何时钟	TRST	—	—	**3**
非法操作码	任何时钟	TRST	—	—	**3**
未初始化的 W 寄存器	任何时钟	TRST	—	—	**3**
陷阱冲突	任何时钟	TRST	—	—	**3**

注　1： TPOR = 上电复位延时 （标称值为 10 μs）。

2： TSTARTUP = 标称值为 20 μs 的条件 POR 延时 （如果使能片上稳压器）或标称值为 64 ms 的上电延时定时器延时 （如果禁止稳压器）。只有在使能稳压器时，所有从掉电状态返回的情况 （包括从休眠模式唤醒），都要应用 TSTARTUP 延时。

3： TRST = 内部状态复位时间 （标称值为 20 μs）。

4： TOST = 振荡器起振定时器延时。10 位计数器计数 1024 个振荡器周期后，才将振荡器时钟释放给系统使用。

5： TLOCK = PLL 锁定时间 （标称值为 20 μs）。

6： TFSCM = 故障保护时钟监视器延时 （标称值为 100 μs）。

5.2.1　　POR 和长振荡器起振时间

振荡器起振电路及其相关的延时定时器与上电时发生的器件复位延时无关。某些晶振电路 （尤其是低频晶振）的起振时间会相对较长。因此，在 SYSRST 信号发出后，可能会发生以下一种或多种情况：

- 振荡电路尚未起振。
- 振荡器起振定时器尚未超时 （如果使用了晶振）。
- PLL 未实现锁定 （如果使用了 PLL）。

在有效时钟源供系统使用前，器件不会开始执行代码。因此，如果必须确定复位延时，必须考虑到振荡器和 PLL 起振延时。

5.2.2　　故障保护时钟监视器 （FSCM）和器件复位

如果使能了 FSCM，它将在发出 $\overline{\text{SYSRST}}$ 信号时开始监视系统时钟源。如果有效时钟源在此时不可用，器件会自动切换至 FRC 振荡器，用户可以切换至陷阱服务程序中要求的晶振。

5.2.2.1　　晶振和 PLL 时钟源的 FSCM 延时

当系统时钟源由晶振和 / 或 PLL 提供时，在 POR 和 PWRT 延时后会自动插入一小段延时 TFSCM。在此延时结束前，FSCM 不会开始监视系统时钟源。FSCM 延时的标称值为 500 μs，为振荡器和 / 或 PLL 稳定下来提供了更多的时间。在大多数情况下，如果禁止了 PWRT，FSCM 延时会防止在器件复位时产生振荡器故障陷阱。

5.3 特殊功能寄存器的复位状态

大多数与 CPU 和外设相关的特殊功能寄存器（SFR）会在器件复位时复位为某个特定值。 SFR 是按其外设或 CPU 功能分组的，其复位值在本手册的相应章节中说明。

除了两个寄存器外，所有其他SFR的复位值都与复位类型无关。复位控制寄存器RCON的复位值取决于器件复位的类型。振荡器控制寄存器 OSCCON 的复位值取决于复位类型和FOSC配置寄存器中对振荡器配置位设置的值。

© 2007 Microchip Technology Inc.

6.0 中断控制器

注：	本数据手册总结了dsPIC33FJXXXGPX06/X08/X10 器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

dsPIC33FJXXXGPX06/X08/X10 中断控制器将诸多外设中断请求信号缩减到一个到 dsPIC33FJXXXGPX06/X08/X10 CPU 的中断请求。该控制器具有以下特性：

- 多达 8 个处理器异常和软件陷阱
- 7 个可由用户选择的优先级
- 具有多达 118 个向量的中断向量表 （Interrupt Vector Table， IVT）
- 每个中断或异常源对应一个唯一的向量
- 在所指定的用户优先级内具有固定的优先级
- 用于支持调试的备用中断向量表 （Alternate Interrupt Vector Table， AIVT）
- 固定的中断进入和返回延时

6.1 中断向量表

中断向量表如图 6-1 所示。 IVT 位于程序存储器中，起始存储单元地址是 000004h。IVT 包含 126 个向量，由 8 个不可屏蔽的陷阱向量和多达 118 个中断源组成。一般来说，每个中断源都有自己的中断向量。每个中断向量都包含一个 24 位宽的地址。每个中断向量存储单元中设置的值是其相关的中断服务程序 （ISR）的起始地址。

中断向量有一个自然优先级；也就是说每个中断向量的优先级与其在向量表中的位置有关。如果其他方面都相同，较低地址的中断向量具有较高的自然优先级。例如，与向量 0 相关的中断比任何其他向量地址的中断具有更高的自然优先级。

dsPIC33FJXXXGPX06/X08/X10 器件实现了多达 67 个唯一中断和 5 个不可屏蔽的陷阱。表 6-1 和表 6-2 对此做了总结。

6.1.1 备用向量表

备用中断向量表 （AIVT）位于 IVT 之后，如图 6-1 所示。ALTIVT 控制位（INTCON2<15>）控制对 AIVT 的访问。如果 ALTIVT 位置 1，则所有的中断和异常处理都将使用备用向量，而非默认向量。备用向量与默认向量的组织方式相同。

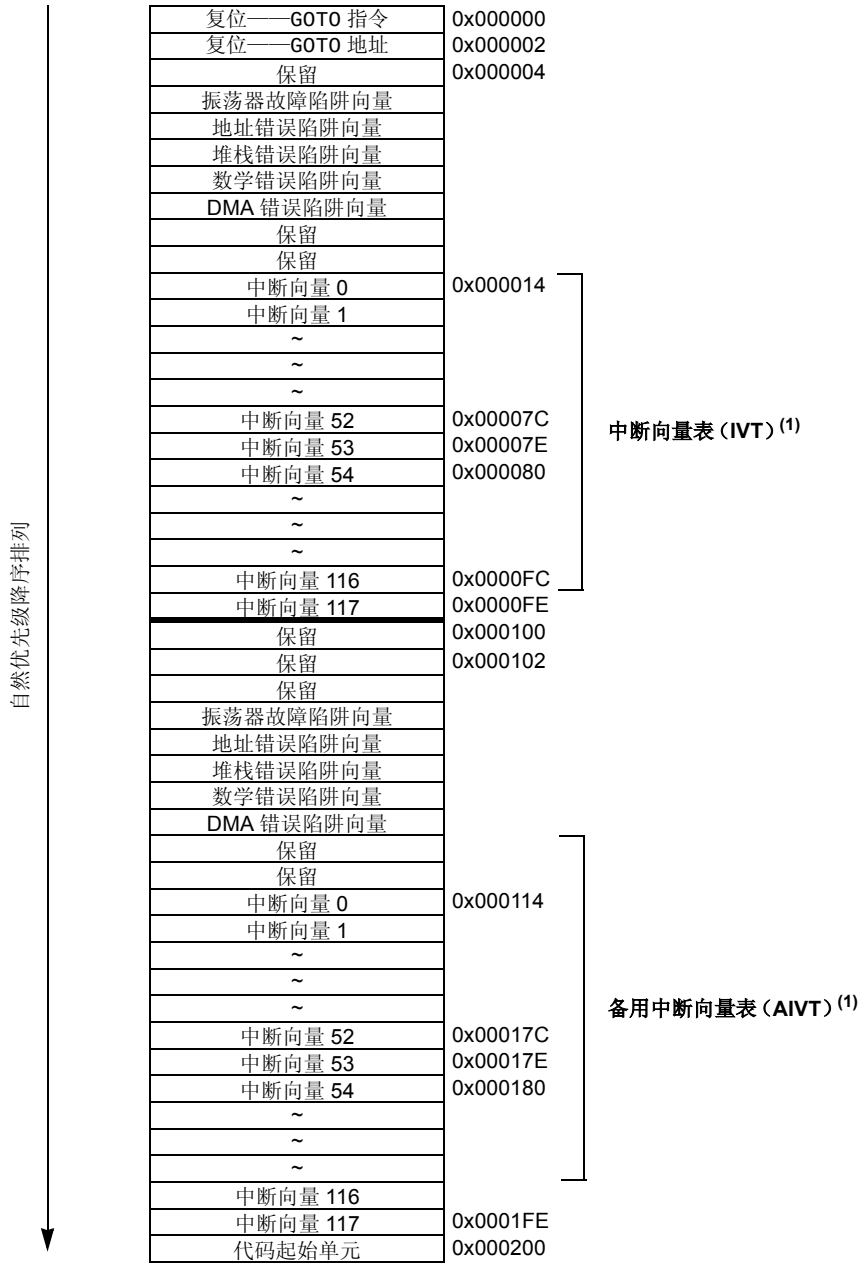
AIVT支持调试功能，它提供了一种不需要将中断向量再编程就可以在应用程序和支持环境之间切换的方法。此特性也支持运行时在不同应用程序之间切换以便评估各种软件算法。如果不需要 AIVT，则应该用 IVT 中使用的地址设置 AIVT。

6.2 复位过程

器件复位不是真正的异常，因为复位过程中并不涉及到中断控制器。作为对复位的响应，dsPIC33FJXXXGPX06/X08/X10 器件清零其寄存器，同时强制 PC 为零。然后数字信号控制器从地址 0x000000 处开始执行程序。用户在复位地址处设定一条 GOTO 指令，该指令会将程序重定向到相应的启动程序。

注：	应该使用包含 RESET 指令的默认中断处理程序的入口地址编程 IVT 和 AIVT 中所有未实现或未使用的向量存储单元。

图 6-1：　　dsPIC33FJXXXGPX06/X08/X10 中断向量表



注　1： 请参见表 6-1 了解所实现的中断向量列表。

© 2007 Microchip Technology Inc.

表 6-1：　　中断向量

向量编号	中断请求（IRQ）编号	IVT 地址	AIVT 地址	中断源
8	0	0x000014	0x000114	INT0——外部中断 0
9	1	0x000016	0x000116	IC1——输入捕捉 1
10	2	0x000018	0x000118	OC1——输出比较 1
11	3	0x00001A	0x00011A	T1——Timer1
12	4	0x00001C	0x00011C	DMA0——DMA 通道 0
13	5	0x00001E	0x00011E	IC2——输入捕捉 2
14	6	0x000020	0x000120	OC2——输出比较 2
15	7	0x000022	0x000122	T2——Timer2
16	8	0x000024	0x000124	T3——Timer3
17	9	0x000026	0x000126	SPI1E——SPI1 错误
18	10	0x000028	0x000128	SPI1——SPI1 传输完成
19	11	0x00002A	0x00012A	U1RX——UART1 接收器
20	12	0x00002C	0x00012C	U1TX——UART1 发送器
21	13	0x00002E	0x00012E	ADC1——ADC 1
22	14	0x000030	0x000130	DMA1——DMA 通道 1
23	15	0x000032	0x000132	保留
24	16	0x000034	0x000134	SI2C1——I2C1 从事件
25	17	0x000036	0x000136	MI2C1——I2C1 主事件
26	18	0x000038	0x000138	保留
27	19	0x00003A	0x00013A	电平变化通知中断
28	20	0x00003C	0x00013C	INT1——外部中断 1
29	21	0x00003E	0x00013E	ADC2——ADC 2
30	22	0x000040	0x000140	IC7——输入捕捉 7
31	23	0x000042	0x000142	IC8——输入捕捉 8
32	24	0x000044	0x000144	DMA2——DMA 通道 2
33	25	0x000046	0x000146	OC3——输出比较 3
34	26	0x000048	0x000148	OC4——输出比较 4
35	27	0x00004A	0x00014A	T4——Timer4
36	28	0x00004C	0x00014C	T5——Timer5
37	29	0x00004E	0x00014E	INT2——外部中断 2
38	30	0x000050	0x000150	U2RX——UART2 接收器
39	31	0x000052	0x000152	U2TX——UART2 发送器
40	32	0x000054	0x000154	SPI2E——SPI2 错误
41	33	0x000056	0x000156	SPI1——SPI1 传输完成
42	34	0x000058	0x000158	C1RX——ECAN1 接收数据就绪
43	35	0x00005A	0x00015A	C1——ECAN1 事件
44	36	0x00005C	0x00015C	DMA3——DMA 通道 3
45	37	0x00005E	0x00015E	IC3——输入捕捉 3
46	38	0x000060	0x000160	IC4——输入捕捉 4
47	39	0x000062	0x000162	IC5——输入捕捉 5
48	40	0x000064	0x000164	IC6——输入捕捉 6
49	41	0x000066	0x000166	OC5——输出比较 5
50	42	0x000068	0x000168	OC6——输出比较 6
51	43	0x00006A	0x00016A	OC7——输出比较 7
52	44	0x00006C	0x00016C	OC8——输出比较 8
53	45	0x00006E	0x00016E	保留

表 6-1： 中断向量 （续）

向量编号	中断请求（IRQ）编号	IVT 地址	AIVT 地址	中断源
54	46	0x000070	0x000170	DMA4——DMA 通道 4
55	47	0x000072	0x000172	T6——Timer6
56	48	0x000074	0x000174	T7——Timer7
57	49	0x000076	0x000176	SI2C2——I2C2 从事件
58	50	0x000078	0x000178	MI2C2——I2C2 主事件
59	51	0x00007A	0x00017A	T8——Timer8
60	52	0x00007C	0x00017C	T9——Timer9
61	53	0x00007E	0x00017E	INT3——外部中断 3
62	54	0x000080	0x000180	INT4——外部中断 4
63	55	0x000082	0x000182	C2RX——ECAN2 接收数据就绪
64	56	0x000084	0x000184	C2——ECAN2 事件
65	57	0x000086	0x000186	保留
66	58	0x000088	0x000188	保留
67	59	0x00008A	0x00018A	DCIE——DCI 错误
68	60	0x00008C	0x00018C	DCID——DCI 传输完成
69	61	0x00008E	0x00018E	DMA5——DMA 通道 5
70	62	0x000090	0x000190	保留
71	63	0x000092	0x000192	保留
72	64	0x000094	0x000194	保留
73	65	0x000096	0x000196	U1E——UART1 错误
74	66	0x000098	0x000198	U2E——UART2 错误
75	67	0x00009A	0x00019A	保留
76	68	0x00009C	0x00019C	DMA6——DMA 通道 6
77	69	0x00009E	0x00019E	DMA7——DMA 通道 7
78	70	0x0000A0	0x0001A0	C1TX——ECAN1 发送数据请求
79	71	0x0000A2	0x0001A2	C2TX——ECAN2 发送数据请求
80-125	72-117	0x0000A4-0x0000FE	0x0001A4-0x0001FE	保留

表 6-2： 陷阱向量

向量编号	IVT 地址	AIVT 地址	陷阱源
0	0x000004	0x000104	保留
1	0x000006	0x000106	振荡器故障
2	0x000008	0x000108	地址错误
3	0x00000A	0x00010A	堆栈错误
4	0x00000C	0x00010C	数学错误
5	0x00000E	0x00010E	DMA 错误陷阱
6	0x000010	0x000110	保留
7	0x000012	0x000112	保留

© 2007 Microchip Technology Inc.

6.3 中断控制和状态寄存器

dsPIC33FJXXXGPX06/X08/X10 器件共有 30 个用于中断控制器的寄存器：

- INTCON1
- INTCON2
- IFS0 至 IFS4
- IEC0 至 IEC4
- IPC0 至 IPC17
- INTTREG

INTCON1 和 INTCON2 控制全局中断。INTCON1 中含有中断嵌套禁止（NSTDIS）位，以及处理器陷阱源的控制和状态标志。 INTCON2 寄存器控制外部中断请求信号行为以及备用中断向量表的使用。

IFS 寄存器包含所有中断请求标志。每个中断源都具有一个状态位，该状态位由相应的外设或外部信号置 1，通过软件进行清零。

IEC 寄存器包含所有中断允许位。这些控制位用于单独允许外设或外部信号中断。

IPC 寄存器用于设置每个中断源的中断优先级。可以给每个用户中断源分配 8 个优先级之一。

INTTREG 寄存器包含相关的中断向量编号和新的 CPU 中断优先级，分别锁存在 INTTREG 寄存器中的向量编号（VECNUM<6:0>）和中断优先级（ILR<3:0>）位域中。新的中断优先级是等待处理中断的优先级。

中断源按表 6-1 中的顺序分配给 IFSx、IECx 和 IPCx 寄存器。例如， INT0 （外部中断 0）表示向量编号为 8，自然优先级为 0 的外部中断。所以 INT0IF 位在 IFS0<0> 中， INT0IE 位在 IEC0<0> 中， INT0IP 位在 IPC0 的第一个位位置 （IPC0<2:0>）中。

尽管两个 CPU 控制寄存器不是中断控制硬件的特定组成部分，但它们仍包含控制中断功能的位。CPU 状态寄存器 SR 包含 IPL<2:0> 位（SR<7:5>）。这些位表示当前 CPU 中断优先级。用户可以通过写 IPL 位来更改当前 CPU 优先级。

CORCON 寄存器包含 IPL3 位，这个位与 IPL<2:0> 位一起表示当前 CPU 优先级。 IPL3 是只读位，所以用户软件无法屏蔽陷阱事件。

在下面各页中的寄存器 6-1 到寄存器 6-32 说明了所有的中断寄存器。

寄存器 6-1：　　SR：CPU 状态寄存器 (1)

R-0	R-0	R/C-0	R/C-0	R-0	R/C-0	R -0	R/W-0
OA	OB	SA	SB	OAB	SAB	DA	DC
bit 15							bit 8

R/W-0(3)	R/W-0(3)	R/W-0(3)	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL2(2)	**IPL1(2)**	**IPL0(2)**	RA	N	OV	Z	C
bit 7							bit 0

图注：		
C = 只可清零位	R = 可读位	U = 未实现位，读为 0
S = 只可置 1 位	W = 可写位	-n = POR 时的值
1 = 置 1	0 = 清零	x = 未知

bit 7-5　　**IPL<2:0>**：CPU 中断优先级状态位 (1)

111 = CPU 中断优先级为 7 （15），禁止用户中断
110 = CPU 中断优先级为 6 （14）
101 = CPU 中断优先级为 5 （13）
100 = CPU 中断优先级为 4 （12）
011 = CPU 中断优先级为 3 （11）
010 = CPU 中断优先级为 2 （10）
001 = CPU 中断优先级为 1 （9）
000 = CPU 中断优先级为 0 （8）

注　1： 如需了解整个寄存器的详细信息，请参见**寄存器 2-1：“SR：CPU 状态寄存器”**。

2： IPL<2:0> 位与 IPL<3> 位 （CORCON<3>）组合形成 CPU 中断优先级。如果 IPL<3> = 1，那么括号中的值表示 IPL。当 IPL<3> = 1 时，禁止用户中断。

3： 当 NSTDIS （INTCON1<15>） = 1 时， IPL<2:0> 状态位是只读的。

寄存器 6-2：　　CORCON：内核控制寄存器 (1)

U-0	U-0	U-0	R/W-0	R/W-0	R-0	R-0	R-0
—	—	—	US	EDT	DL<2:0>		
bit 15							bit 8

R/W-0	R/W-0	R/W-1	R/W-0	R/C-0	R/W-0	R/W-0	R/W-0
SATA	SATB	SATDW	ACCSAT	**IPL3(2)**	PSV	RND	IF
bit 7							bit 0

图注：	C = 只可清零位		
R = 可读位	W = 可写位	-n = POR 时的值	1 = 置 1
0 = 清零	x = 未知	U = 未实现位，读为 0	

bit 3　　**IPL3**：CPU 中断优先级状态位 3(2)

1 = CPU 中断优先级大于 7
0 = CPU 中断优先级等于或小于 7

注　1： 如需了解整个寄存器的详细信息，请参见**寄存器 2-2：“CORCON：内核控制寄存器”**。

2： IPL3 位与 IPL<2:0> 位 （SR<7:5>）组合形成 CPU 中断优先级。

© 2007 Microchip Technology Inc.

寄存器 6-3：　　INTCON1：中断控制寄存器 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NSTDIS	OVAERR	OVBERR	COVAERR	COVBERR	OVATE	OVBTE	COVTE
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
SFTACERR	DIV0ERR	DMACERR	MATHERR	ADDRERR	STKERR	OSCFAIL	—
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**NSTDIS：**中断嵌套禁止位
1 = 禁止中断嵌套
0 = 使能中断嵌套

bit 14　**OVAERR：**累加器 A 溢出陷阱标志位
1 = 陷阱由累加器 A 溢出引起
0 = 陷阱不是由累加器 A 溢出引起

bit 13　**OVBERR：**累加器 B 溢出陷阱标志位
1 = 陷阱由累加器 B 溢出引起
0 = 陷阱不是由累加器 B 溢出引起

bit 12　**COVAERR：**累加器 A 灾难性溢出陷阱标志位
1 = 陷阱由累加器 A 灾难性溢出引起
0 = 陷阱不是由累加器 A 灾难性溢出引起

bit 11　**COVBERR：**累加器 B 灾难性溢出陷阱标志位
1 = 陷阱由累加器 B 灾难性溢出引起
0 = 陷阱不是由累加器 B 灾难性溢出引起

bit 10　**OVATE：**累加器 A 溢出陷阱允许位
1 = 允许累加器 A 溢出陷阱
0 = 禁止陷阱

bit 9　**OVBTE：**累加器 B 溢出陷阱允许位
1 = 允许累加器 B 溢出陷阱
0 = 禁止陷阱

bit 8　**COVTE：**灾难性溢出陷阱允许位
1 = 允许累加器 A 或 B 的灾难性溢出陷阱
0 = 禁止陷阱

bit 7　**SFTACERR：**累加器移位错误状态位
1 = 数学错误陷阱由非法累加器移位引起
0 = 数学错误陷阱不是由非法累加器移位引起

bit 6　**DIV0ERR：**算术错误状态位
1 = 数学错误陷阱由被零除引起
0 = 数学错误陷阱不是由被零除引起

bit 5　**DMACERR：**DMA 控制器错误状态位
1 = 发生了 DMA 控制器错误陷阱
0 = 未发生 DMA 控制器错误陷阱

bit 4　**MATHERR：**算术错误状态位
1 = 发生了数学错误陷阱
0 = 未发生数学错误陷阱

寄存器 6-3：　　INTCON1：中断控制寄存器 1 （续）

bit 3　　**ADDRERR：**地址错误陷阱状态位
1 = 发生了地址错误陷阱
0 = 未发生地址错误陷阱

bit 2　　**STKERR：**堆栈错误陷阱状态位
1 = 发生了堆栈错误陷阱
0 = 未发生堆栈错误陷阱

bit 1　　**OSCFAIL：**振荡器故障陷阱状态位
1 = 发生了振荡器故障陷阱
0 = 未发生振荡器故障陷阱

bit 0　　**未实现：**读为 0

© 2007 Microchip Technology Inc.

寄存器 6-4：　　INTCON2：中断控制寄存器 2

R/W-0	R-0	U-0	U-0	U-0	U-0	U-0	U-0
ALTIVT	DISI	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	INT4EP	INT3EP	INT2EP	INT1EP	INT0EP
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**ALTIVT**：备用中断向量表使能位
1 = 使用备用向量表
0 = 使用标准 （默认）向量表

bit 14　**DISI**：DISI 指令状态位
1 = 执行了 DISI 指令
0 = 没有执行 DISI 指令

bit 13-5　**未实现**：读为 0

bit 4　**INT4EP**：外部中断 4 边沿检测极性选择位
1 = 下降沿中断
0 = 上升沿中断

bit 3　**INT3EP**：外部中断 3 边沿检测极性选择位
1 = 下降沿中断
0 = 上升沿中断

bit 2　**INT2EP**：外部中断 2 边沿检测极性选择位
1 = 下降沿中断
0 = 上升沿中断

bit 1　**INT1EP**：外部中断 1 边沿检测极性选择位
1 = 下降沿中断
0 = 上升沿中断

bit 0　**INT0EP**：外部中断 0 边沿检测极性选择位
1 = 下降沿中断
0 = 上升沿中断

寄存器 6-5：　　IFS0：中断标志状态寄存器 0

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	DMA1IF	AD1IF	U1TXIF	U1RXIF	SPI1IF	SPI1EIF	T3IF
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T2IF	OC2IF	IC2IF	DMA01IF	T1IF	OC1IF	IC1IF	INT0IF
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**未实现：**读为 0

bit 14　**DMA1IF：**DMA 通道 1 数据传输完成中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 13　**AD1IF：**ADC1 转换完成中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 12　**U1TXIF：**UART1 发送器中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 11　**U1RXIF：**UART1 接收器中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 10　**SPI1IF：**SPI1 事件中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 9　**SPI1EIF：**SPI1 故障中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 8　**T3IF：**Timer3 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 7　**T2IF：**Timer2 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 6　**OC2IF：**输出比较通道 2 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 5　**IC2IF：**输入捕捉通道 2 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 4　**DMA0IF：**DMA 通道 0 数据传输完成中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 3　**T1IF：**Timer1 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

© 2007 Microchip Technology Inc.

寄存器 6-5：　　IFS0：中断标志状态寄存器 0 （续）

bit 2　　**OC1IF**：输出比较通道 1 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 1　　**IC1IF**：输入捕捉通道 1 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 0　　**INT0IF**：外部中断 0 标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

寄存器 6-6：　　IFS1：中断标志状态寄存器 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	DMA21IF
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IC8IF	IC7IF	AD2IF	INT1IF	CNIF	—	MI2C1IF	SI2C1IF
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**U2TXIF**：UART2 发送器中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 14　**U2RXIF**：UART2 接收器中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 13　**INT2IF**：外部中断 2 标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 12　**T5IF**：Timer5 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 11　**T4IF**：Timer4 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 10　**OC4IF**：输出比较通道 4 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 9　**OC3IF**：输出比较通道 3 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 8　**DMA2IF**：DMA 通道 2 数据传输完成中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 7　**IC8IF**：输入捕捉通道 8 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 6　**IC7IF**：输入捕捉通道 7 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 5　**AD2IF**：ADC2 转换完成中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 4　**INT1IF**：外部中断 1 标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

© 2007 Microchip Technology Inc.

寄存器 6-6：　　IFS1：中断标志状态寄存器 1 （续）

bit 3	**CNIF**：输入电平变化通知中断标志状态位 1 = 产生了中断请求 0 = 未产生中断请求
bit 2	**未实现**：读为 0
bit 1	**MI2C1IF**：I2C1 主事件中断标志状态位 1 = 产生了中断请求 0 = 未产生中断请求
bit 0	**SI2C1IF**：I2C1 从事件中断标志状态位 1 = 产生了中断请求 0 = 未产生中断请求

寄存器 6-7：　　IFS2：中断标志状态寄存器 2

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T6IF	DMA4IF	—	OC8IF	OC7IF	OC6IF	OC5IF	IC6IF
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IC5IF	IC4IF	IC3IF	DMA3IF	C1IF	C1RXIF	SPI2IF	SPI2EIF
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**T6IF**：Timer6 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 14　**DMA4IF**：DMA 通道 4 数据传输完成中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 13　**未实现**：读为 0

bit 12　**OC8IF**：输出比较通道 8 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 11　**OC7IF**：输出比较通道 7 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 10　**OC6IF**：输出比较通道 6 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 9　**OC5IF**：输出比较通道 5 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 8　**IC6IF**：输入捕捉通道 6 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 7　**IC5IF**：输入捕捉通道 5 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 6　**IC4IF**：输入捕捉通道 4 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 5　**IC3IF**：输入捕捉通道 3 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 4　**DMA3IF**：DMA 通道 3 数据传输完成中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 3　**C1IF**：ECAN1 事件中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

© 2007 Microchip Technology Inc.

寄存器 6-7：　　IFS2：中断标志状态寄存器 2 （续）

bit 2　　**C1RXIF**：ECAN1 接收数据就绪中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 1　　**SPI2IF**：SPI2 事件中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 0　　**SPI2EIF**：SPI2 错误中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

寄存器 6-8：　　IFS3：中断标志状态寄存器 3

U-0	U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0
—	—	DMA5IF	DCIIF	DCIEIF	—	—	C2IF
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
C2RXIF	INT4IF	INT3IF	T9IF	T8IF	MI2C2IF	SI2C2IF	T7IF
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-14　**未实现：**读为 0

bit 13　**DMA5IF：**DMA 通道 5 数据传输完成中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 12　**DCIIF：**DCI 事件中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 11　**DCIEIF：**DCI 错误中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 10-9　**未实现：**读为 0

bit 8　**C2IF：**ECAN2 事件中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 7　**C2RXIF：**ECAN2 接收数据就绪中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 6　**INT4IF：**外部中断 4 标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 5　**INT3IF：**外部中断 3 标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 4　**T9IF：**Timer9 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 3　**T8IF：**Timer8 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 2　**MI2C2IF：**I2C2 主事件中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 1　**SI2C2IF：**I2C2 从事件中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 0　**T7IF：**Timer7 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

© 2007 Microchip Technology Inc.

寄存器 6-9：　　IFS4：中断标志状态寄存器 4

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0
C2TXIF	C1TXIF	DMA7IF	DMA6IF	—	U2EIF	U1EIF	—
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-8　**未实现：**读为 0

bit 7　**C2TXIF：**ECAN2 发送数据请求中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 6　**C1TXIF：**ECAN1 发送数据请求中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 5　**DMA7IF：**DMA 通道 7 数据传输完成中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 4　**DMA6IF：**DMA 通道 6 数据传输完成中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 3　**未实现：**读为 0

bit 2　**U2EIF：**UART2 错误中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 1　**U1EIF：**UART1 错误中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 0　**未实现：**读为 0

寄存器 6-10：　IEC0：中断允许控制寄存器 0

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	DMA1IE	AD1IE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T2IE	OC2IE	IC2IE	DMA0IE	T1IE	OC1IE	IC1IE	INT0IE
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**未实现**：读为 0

bit 14　**DMA1IE**：DMA 通道 1 数据传输完成中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 13　**AD1IE**：ADC1 转换完成中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 12　**U1TXIE**：UART1 发送器中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 11　**U1RXIE**：UART1 接收器中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 10　**SPI1IE**：SPI1 事件中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 9　**SPI1EIE**：SPI1 错误中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 8　**T3IE**：Timer3 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 7　**T2IE**：Timer2 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 6　**OC2IE**：输出比较通道 2 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 5　**IC2IE**：输入捕捉通道 2 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 4　**DMA0IE**：DMA 通道 0 数据传输完成中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 3　**T1IE**：Timer1 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

© 2007 Microchip Technology Inc.

寄存器 6-10：　　IEC0：中断允许控制寄存器 0 （续）

bit 2　　**OC1IE：**输出比较通道 1 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 1　　**IC1IE：**输入捕捉通道 1 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 0　　**INT0IE：**外部中断 0 允许位
1 = 允许中断请求
0 = 禁止中断请求

寄存器 6-11： IEC1：中断允许控制寄存器 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	DMA2IE
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IC8IE	IC7IE	AD2IE	INT1IE	CNIE	—	MI2C1IE	SI2C1IE
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **U2TXIE**：UART2 发送器中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 14 **U2RXIE**：UART2 接收器中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 13 **INT2IE**：外部中断 2 允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 12 **T5IE**：Timer5 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 11 **T4IE**：Timer4 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 10 **OC4IE**：输出比较通道 4 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 9 **OC3IE**：输出比较通道 3 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 8 **DMA2IE**：DMA 通道 2 数据传输完成中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 7 **IC8IE**：输入捕捉通道 8 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 6 **IC7IE**：输入捕捉通道 7 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 5 **AD2IE**：ADC2 转换完成中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 4 **INT1IE**：外部中断 1 允许位
1 = 允许中断请求
0 = 禁止中断请求

© 2007 Microchip Technology Inc.

寄存器 6-11：　　IEC1：中断允许控制寄存器 1 （续）

bit 3　　**CNIE**：输入电平变化通知中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 2　　**未实现**：读为 0

bit 1　　**MI2C1IE**：I2C1 主事件中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 0　　**SI2C1IE**：I2C1 从事件中断允许位
1 = 允许中断请求
0 = 禁止中断请求

寄存器 6-12：　IEC2：中断允许控制寄存器 2

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T6IE	DMA4IE	—	OC8IE	OC7IE	OC6IE	OC5IE	IC6IE
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IC5IE	IC4IE	IC3IE	DMA3IE	C1IE	C1RXIE	SPI2IE	SPI2EIE
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**T6IE**：Timer6 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 14　**DMA4IE**：DMA 通道 4 数据传输完成中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 13　**未实现**：读为 0

bit 12　**OC8IE**：输出比较通道 8 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 11　**OC7IE**：输出比较通道 7 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 10　**OC6IE**：输出比较通道 6 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 9　**OC5IE**：输出比较通道 5 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 8　**IC6IE**：输入捕捉通道 6 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 7　**IC5IE**：输入捕捉通道 5 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 6　**IC4IE**：输入捕捉通道 4 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 5　**IC3IE**：输入捕捉通道 3 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 4　**DMA3IE**：DMA 通道 3 数据传输完成中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 3　**C1IE**：ECAN1 事件中断允许位
1 = 允许中断请求
0 = 禁止中断请求

© 2007 Microchip Technology Inc.

寄存器 6-12： IEC2：中断允许控制寄存器 2 （续）

bit 2 **C1RXIE**：ECAN1 接收数据就绪中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 1 **SPI2IE**：SPI2 事件中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 0 **SPI2EIE**：SPI2 错误中断允许位
1 = 允许中断请求
0 = 禁止中断请求

寄存器 6-13：　　IEC3：中断允许控制寄存器 3

U-0	U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0
—	—	DMA5IE	DCIIE	DCIEIE	—	—	C2IE
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
C2RXIE	INT4IE	INT3IE	T9IE	T8IE	MI2C2IE	SI2C2IE	T7IE
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-14　**未实现：**读为 0

bit 13　**DMA5IE：**DMA 通道 5 数据传输完成中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 12　**DCIIE：**DCI 事件中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 11　**DCIEIE：**DCI 错误中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 10-9　**未实现：**读为 0

bit 8　**C2IE：**ECAN2 事件中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 7　**C2RXIE：**ECAN2 接收数据就绪中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 6　**INT4IE：**外部中断 4 允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 5　**INT3IE：**外部中断 3 允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 4　**T9IE：**Timer9 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 3　**T8IE：**Timer8 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 2　**MI2C2IE：**I2C2 主事件中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 1　**SI2C2IE：**I2C2 从事件中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 0　**T7IE：**Timer7 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

© 2007 Microchip Technology Inc.

寄存器 6-14：　　IEC4：中断允许控制寄存器 4

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0
C2TXIE	C1TXIE	DMA7IE	DMA6IE	—	U2EIE	U1EIE	—
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-8　**未实现：**读为 0

bit 7　**C2TXIE：**ECAN2 发送数据请求中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 6　**C1TXIE：**ECAN1 发送数据请求中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 5　**DMA7IE：**DMA 通道 7 数据传输完成中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 4　**DMA6IE：**DMA 通道 6 数据传输完成中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 3　**未实现：**读为 0

bit 2　**U2EIE：**UART2 错误中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 1　**U1EIE：**UART1 错误中断允许位
1 = 允许中断请求
0 = 禁止中断请求

bit 0　**未实现：**读为 0

寄存器 6-15：　　IPC0：中断优先级控制寄存器 0

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T1IP<2:0>			—	OC1IP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	IC1IP<2:0>			—	INT0IP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**未实现：**读为 0

bit 14-12　**T1IP<2:0>：**Timer1 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 11　**未实现：**读为 0

bit 10-8　**OC1IP<2:0>：**输出比较通道 1 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 7　**未实现：**读为 0

bit 6-4　**IC1IP<2:0>：**输入捕捉通道 1 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3　**未实现：**读为 0

bit 2-0　**INT0IP<2:0>：**外部中断 0 优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

© 2007 Microchip Technology Inc.

寄存器 6-16：　　IPC1：中断优先级控制寄存器 1

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T2IP<2:0>			—	OC2IP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	IC2IP<2:0>			—	DMA0IP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**未实现：**读为 0

bit 14-12　**T2IP<2:0>：**Timer2 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 11　**未实现：**读为 0

bit 10-8　**OC2IP<2:0>：**输出比较通道 2 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 7　**未实现：**读为 0

bit 6-4　**IC2IP<2:0>：**输入捕捉通道 2 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3　**未实现：**读为 0

bit 2-0　**DMA0IP<2:0>：**DMA 通道 0 数据传输完成中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

寄存器 6-17：　　IPC2：中断优先级控制寄存器 2

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	U1RXIP<2:0>			—	SPI1IP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	SPI1EIP<2:0>			—	T3IP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**未实现：**读为 0

bit 14-12　**U1RXIP<2:0>：**UART1 接收器中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 11　**未实现：**读为 0

bit 10-8　**SPI1IP<2:0>：**SPI1 事件中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 7　**未实现：**读为 0

bit 6-4　**SPI1EIP<2:0>：**SPI1 错误中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3　**未实现：**读为 0

bit 2-0　**T3IP<2:0>：**Timer3 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

© 2007 Microchip Technology Inc.

寄存器 6-18：　　IPC3：中断优先级控制寄存器 3

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	DMA1IP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	AD1IP<2:0>			—	U1TXIP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-11　**未实现：**读为 0

bit 10-8　**DMA1IP<2:0>：**DMA 通道 1 数据传输完成中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 7　**未实现：**读为 0

bit 6-4　**AD1IP<2:0>：**ADC1 转换完成中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3　**未实现：**读为 0

bit 2-0　**U1TXIP<2:0>：**UART1 发送器中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

寄存器 6-19：　　IPC4：中断优先级控制寄存器 4

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	CNIP<2:0>			—	—	—	—
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	MI2C1IP<2:0>			—	SI2C1IP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**未实现：**读为 0

bit 14-12　**CNIP<2:0>：**电平变化通知中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 11-7　**未实现：**读为 0

bit 6-4　**MI2C1IP<2:0>：**I2C1 主事件中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3　**未实现：**读为 0

bit 2-0　**SI2C1IP<2:0>：**I2C1 从事件中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

© 2007 Microchip Technology Inc.

寄存器 6-20：　　IPC5：中断优先级控制寄存器 5

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	IC8IP<2:0>			—	IC7IP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	AD2IP<2:0>			—	INT1IP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**未实现：**读为 0

bit 14-12　**IC8IP<2:0>：**输入捕捉通道 8 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 11　**未实现：**读为 0

bit 10-8　**IC7IP<2:0>：**输入捕捉通道 7 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 7　**未实现：**读为 0

bit 6-4　**AD2IP<2:0>：**ADC2 转换完成中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3　**未实现：**读为 0

bit 2-0　**INT1IP<2:0>：**外部中断 1 优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

寄存器 6-21： IPC6：中断优先级控制寄存器 6

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T4IP<2:0>			—	OC4IP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	OC3IP<2:0>			—	DMA2IP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **未实现：**读为 0

bit 14-12 **T4IP<2:0>：**Timer4 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 11 **未实现：**读为 0

bit 10-8 **OC4IP<2:0>：**输出比较通道 4 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 7 **未实现：**读为 0

bit 6-4 **OC3IP<2:0>：**输出比较通道 3 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3 **未实现：**读为 0

bit 2-0 **DMA2IP<2:0>：**DMA 通道 2 数据传输完成中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

© 2007 Microchip Technology Inc.

寄存器 6-22：　　IPC7：中断优先级控制寄存器 7

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	U2TXIP<2:0>			—	U2RXIP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	INT2IP<2:0>			—	T5IP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**未实现：**读为 0

bit 14-12　**U2TXIP<2:0>：**UART2 发送器中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 11　**未实现：**读为 0

bit 10-8　**U2RXIP<2:0>：**UART2 接收器中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 7　**未实现：**读为 0

bit 6-4　**INT2IP<2:0>：**外部中断 2 优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3　**未实现：**读为 0

bit 2-0　**T5IP<2:0>：**Timer5 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

寄存器 6-23：　IPC8：中断优先级控制寄存器 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	C1IP<2:0>			—	C1RXIP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	SPI2IP<2:0>			—	SPI2EIP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**未实现：**读为 0

bit 14-12　**C1IP<2:0>：**ECAN1 事件中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 11　**未实现：**读为 0

bit 10-8　**C1RXIP<2:0>：**ECAN1 接收数据就绪中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 7　**未实现：**读为 0

bit 6-4　**SPI2IP<2:0>：**SPI2 事件中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3　**未实现：**读为 0

bit 2-0　**SPI2EIP<2:0>：**SPI2 错误中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

© 2007 Microchip Technology Inc.

寄存器 6-24：　　IPC9：中断优先级控制寄存器 9

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	IC5IP<2:0>			—	IC4IP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	IC3IP<2:0>			—	DMA3IP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**未实现：**读为 0

bit 14-12　**IC5IP<2:0>：**输入捕捉通道 5 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 11　**未实现：**读为 0

bit 10-8　**IC4IP<2:0>：**输入捕捉通道 4 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 7　**未实现：**读为 0

bit 6-4　**IC3IP<2:0>：**输入捕捉通道 3 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3　**未实现：**读为 0

bit 2-0　**DMA3IP<2:0>：**DMA 通道 3 数据传输完成中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

寄存器 6-25：　IPC10：中断优先级控制寄存器 10

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	OC7IP<2:0>			—	OC6IP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	OC5IP<2:0>			—	IC6IP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**未实现：**读为 0

bit 14-12　**OC7IP<2:0>：**输出比较通道 7 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 11　**未实现：**读为 0

bit 10-8　**OC6IP<2:0>：**输出比较通道 6 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 7　**未实现：**读为 0

bit 6-4　**OC5IP<2:0>：**输出比较通道 5 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3　**未实现：**读为 0

bit 2-0　**IC6IP<2:0>：**输入捕捉通道 6 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

© 2007 Microchip Technology Inc.

寄存器 6-26：　　IPC11：中断优先级控制寄存器 11

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T6IP<2:0>			—	DMA4IP<2:0>		
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	OC8IP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**未实现：**读为 0

bit 14-12　**T6IP<2:0>：**Timer6 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 11　**未实现：**读为 0

bit 10-8　**DMA4IP<2:0>：**DMA 通道 4 数据传输完成中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 7-3　**未实现：**读为 0

bit 2-0　**OC8IP<2:0>：**输出比较通道 8 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

寄存器 6-27：　　IPC12：中断优先级控制寄存器 12

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T8IP<2:0>			—	MI2C2IP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	SI2C2IP<2:0>			—	T7IP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**未实现：**读为 0

bit 14-12　**T8IP<2:0>：**Timer8 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 11　**未实现：**读为 0

bit 10-8　**MI2C2IP<2:0>：**I2C2 主事件中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 7　**未实现：**读为 0

bit 6-4　**SI2C2IP<2:0>：**I2C2 从事件中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3　**未实现：**读为 0

bit 2-0　**T7IP<2:0>：**Timer7 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

© 2007 Microchip Technology Inc.

寄存器 6-28： IPC13：中断优先级控制寄存器 13

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	C2RXIP<2:0>			—	INT4IP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	INT3IP<2:0>			—	T9IP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **未实现：**读为 0

bit 14-12 **C2RXIP<2:0>：**ECAN2 接收数据就绪中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 11 **未实现：**读为 0

bit 10-8 **INT4IP<2:0>：**外部中断 4 优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 7 **未实现：**读为 0

bit 6-4 **INT3IP<2:0>：**外部中断 3 优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3 **未实现：**读为 0

bit 2-0 **T9IP<2:0>：**Timer9 中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

寄存器 6-29：　　IPC14：中断优先级控制寄存器 14

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	DCIEIP<2:0>			—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	C2IP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**未实现：**读为 0

bit 14-12　**DCIEIP<2:0>：**DCI 错误中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 11-8　**未实现：**读为 0

bit 7-3　**未实现：**读为 0

bit 2-0　**C2IP<2:0>：**ECAN2 事件中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

© 2007 Microchip Technology Inc.

寄存器 6-30：　　IPC15：中断优先级控制寄存器 15

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	DMA5IP<2:0>			—	DCIIP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-8　**未实现：**读为 0

bit 7　**未实现：**读为 0

bit 6-4　**DMA5IP<2:0>：**DMA 通道 5 数据传输完成中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3　**未实现：**读为 0

bit 2-0　**DCIIP<2:0>：**DCI 事件中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

寄存器 6-31：　　IPC16：中断优先级控制寄存器 16

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	U2EIP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	U1EIP<2:0>			—	—	—	—
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-11　**未实现：**读为 0

bit 10-8　**U2EIP<2:0>：**UART2 错误中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 7　**未实现：**读为 0

bit 6-4　**U1EIP<2:0>：**UART1 错误中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3-0　**未实现：**读为 0

© 2007 Microchip Technology Inc.

寄存器 6-32： IPC17：中断优先级控制寄存器 17

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	C2TXIP<2:0>			—	C1TXIP<2:0>		
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	DMA7IP<2:0>			—	DMA6IP<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **未实现：**读为 0

bit 14-12 **C2TXIP<2:0>：**ECAN2 发送数据请求中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 11 **未实现：**读为 0

bit 10-8 **C1TXIP<2:0>：**ECAN1 发送数据请求中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 7 **未实现：**读为 0

bit 6-4 **DMA7IP<2:0>：**DMA 通道 7 数据传输完成中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

bit 3 **未实现：**读为 0

bit 2-0 **DMA6IP<2:0>：**DMA 通道 6 数据传输完成中断优先级位
111 = 中断优先级为 7（最高优先级中断）
•
•
•
001 = 中断优先级为 1
000 = 禁止中断源

寄存器 6-33：　INTTREG：中断控制和状态寄存器

R-0	R/W-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	ILR<3:0>			
bit 15							bit 8

U-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
—	VECNUM<6:0>						
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-12　**未实现：**读为 0

bit 11-8　**ILR：**新的 CPU 中断优先级位
1111 = CPU 中断优先级为 15
•
•
•
0001 = CPU 中断优先级为 1
0000 = CPU 中断优先级为 0

bit 7　**未实现：**读为 0

bit 6-0　**VECNUM：**待处理中断向量编号位
0111111 = 待处理中断向量的编号为 135
•
•
•
0000001 = 待处理中断向量的编号为 9
0000000 = 待处理中断向量的编号为 8

© 2007 Microchip Technology Inc.

6.4 中断设置过程

6.4.1 初始化

要配置中断源：

1. 如果不需要嵌套中断，则将 NSTDIS 位（INTCON1<15>）置 1。
2. 通过写相应 IPCx 寄存器中的控制位为中断源选择由用户分配的优先级。优先级将取决于具体的应用和中断源类型。如果不需要多重优先级，则可以将所有允许的中断源的 IPCx 寄存器控制位设定为同一个非零值。

注： 在器件复位时，IPCx 寄存器被初始化，为所有用户中断源分配优先级 4。

3. 将相应 IFSx 寄存器中与外设相关的中断标志状态位清零。
4. 通过将相应 IECx 寄存器中与中断源相关的中断允许控制位置 1 来允许中断源。

6.4.2 中断服务程序

用于声明 ISR 和使用正确向量地址初始化 IVT 的方法取决于编程语言（即，C 语言或汇编语言）和用于开发应用程序的语言开发工具套件。通常，用户必须清零 IFSx 寄存器中与 ISR 处理的中断源相对应的中断标志。否则，在从 ISR 退出后将立即重新进入 ISR。如果 ISR 用汇编语言编码，则必须使用 RETFIE 指令结束 ISR，以便将保存的 PC 值、 SRL 值和原先的 CPU 优先级弹出堆栈。

6.4.3 陷阱服务程序

除了必须清零 INTCON1 寄存器中相应的陷阱状态标志来避免重新进入陷阱服务程序（Trap Service Routine，TSR）之外， TSR 使用与 ISR 类似的方式编写。

6.4.4 中断禁止

所有的用户中断均可以使用以下过程禁止：

1. 使用 PUSH 指令将当前的 SR 值压入软件堆栈。
2. 通过将值 OEh 与 SRL 进行逻辑或运算来强制将 CPU 的优先级设置为 7。

如果要允许用户中断，则可以使用 POP 指令来恢复先前的 SR 值。

请注意，只能禁止优先级小于等于 7 的用户中断。不能禁止陷阱源 （优先级为 8-15）。

使用 DISI 指令可以方便地将优先级为 1-6 的中断禁止一段固定的时间。 DISI 指令不能禁止优先级为 7 的中断源。

注：

© 2007 Microchip Technology Inc.

7.0 直接存储器访问 （DMA）

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

直接存储器访问（Direct Memory Access，DMA）是在外设 SFR（如 UART 接收缓冲区和输入捕捉 1 缓冲区）和 RAM 中缓冲区或存储在 RAM 中的变量间复制数据的非常高效的机制，它极少需要 CPU 干预。在每次外设中断发生时，DMA 控制器能自动复制整块数据，无需用户软件读或写外设特殊功能寄存器（SFR）。DMA 控制器使用专用的总线传输数据，因此，不会占用 CPU 的代码执行周期。要使用 DMA 功能，相应的用户缓冲区或变量必须位于 DMA RAM 中。

表 7-1 中列出了可以使用 DMA 的 dsPIC33FJXXXGPX06/X08/X10 外设以及与之相关的中断请求 （Interrupt Request， IRQ）编号。

表 7-1： 支持 DMA 的外设

外设	IRQ 编号
INT0	0
输入捕捉 1	1
输入捕捉 2	5
输出比较 1	2
输出比较 2	6
Timer2	7
Timer3	8
SPI1	10
SPI2	33
UART1 接收	11
UART1 发送	12
UART2 接收	30
UART2 发送	31
ADC1	13
ADC2	21
DCI	60
ECAN1 接收	34
ECAN1 发送	70
ECAN2 接收	55
ECAN2 发送	71

DMA 控制器具有 8 路相同的数据传输通道。

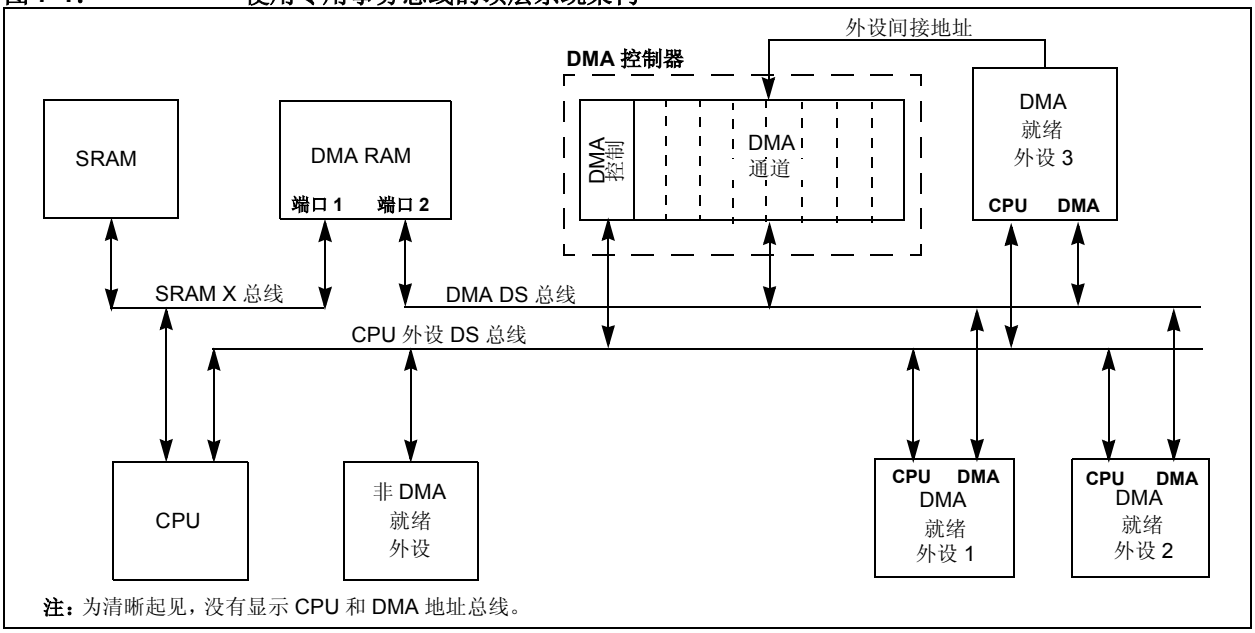
每路通道都有一组控制和状态寄存器。每路 DMA 通道可配置为将数据从双端口 DMA RAM 缓冲区复制到外设 SFR 中，或从外设 SFR 复制到 DMA RAM 缓冲区中。

DMA 控制器支持以下功能：

- 按字或字节传输数据。
- 将数据从外设传输到 DMA RAM 或从 DMA RAM 传输到外设。
- 有或无自动后递增的 DMA RAM 存储单元的间接寻址。
- 外设间接寻址——在某些外设中DMA RAM读/写地址的一部分可能来自外设。
- 单数据块传输——在传输完一个数据块后终止 DMA 传输。
- 连续数据块传输——在完成一次数据块传输后重新装入 DMA RAM 缓冲区起始地址。
- “乒乓”（Ping-Pong）模式——在连续数据块传输之间切换两个 DMA RAM 的起始地址，然后交替填充两个缓冲区。
- 自动或手动启动数据块传输
- 每路通道可以从 20 个可能的数据源或目标源中选择。

对于每路DMA通道，在传输完每块数据后产生一个DMA中断请求。也可在填充完整块数据的一半后产生中断。

图 7-1：　　使用专用事务总线的顶层系统架构



注：为清晰起见，没有显示 CPU 和 DMA 地址总线。

7.1　DMAC 寄存器

每路 DMAC 通道 x（x = 0、1、2、3、4、5、6 或 7）均包含以下寄存器：

- 16 位 DMA 通道控制寄存器 （DMAxCON）
- 16 位 DMA 通道 IRQ 选择寄存器 （DMAxREQ）
- 16 位 DMA RAM 主起始地址寄存器（DMAxSTA）
- 16 位 DMA RAM 辅助起始地址寄存器（DMAxSTB）
- 16 位 DMA 外设地址寄存器 （DMAxPAD）
- 10 位 DMA 传输计数寄存器 （DMAxCNT）

还有一对状态寄存器 DMACS0 和 DMACS1 是所有 DMAC 通道共用的。

© 2007 Microchip Technology Inc.

寄存器 7-1：　　DMAxCON：DMA 通道 x 控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
CHEN	SIZE	DIR	HALF	NULLW	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
—	—	AMODE<1:0>		—	—	MODE<1:0>	
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**CHEN**：通道使能位
1 = 使能通道
0 = 禁止通道

bit 14　**SIZE**：数据传输大小位
1 = 字节
0 = 字

bit 13　**DIR**：传输方向位 （源 / 目标总线选择）
1 = 从 DMA RAM 地址读取，写入外设地址
0 = 从外设地址读取，写入 DMA RAM 地址

bit 12　**HALF**：数据块传输完成中断选择位
1 = 当传送了一半数据时，发出数据块传输完成中断
0 = 当传送了所有数据时，发出数据块传输完成中断

bit 11　**NULLW**：空数据外设写模式选择位
1 = 除将外设 SFR 中的数据写入 DMA RAM 外，还将空数据写入外设 SFR （DIR 位也必须清零）
0 = 正常工作

bit 10-6　**未实现**：读为 0

bit 5-4　**AMODE<1:0>**：DMA 通道寻址模式选择位
11 = 保留
10 = 外设间接寻址模式
01 = 无后递增的寄存器间接寻址模式
00 = 带后递增的寄存器间接寻址模式

bit 3-2　**未实现**：读为 0

bit 1-0　**MODE<1:0>**：DMA 通道工作模式选择位
11 = 使能单数据块乒乓模式 （与每个 DMA RAM 缓冲区之间传输一块数据）
10 = 使能连续数据块乒乓模式
01 = 禁止单数据块乒乓模式
00 = 禁止连续数据块乒乓模式

寄存器 7-2： DMAxREQ：DMA 通道 x IRQ 选择寄存器

R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
FORCE(1)	—	—	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
—	IRQSEL6(2)	IRQSEL5(2)	IRQSEL4(2)	IRQSEL3(2)	IRQSEL2(2)	IRQSEL1(2)	IRQSEL0(2)
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **FORCE**：强制 DMA 传输位 (1)
1 = 强制进行单次 DMA 传输 （手动模式）
0 = 自动按照 DMA 请求进行 DMA 传输

bit 14-7 **未实现**：读为 0

bit 6-0 **IRQSEL<6:0>**：DMA 外设 IRQ 编号选择位 (2)
0000000-1111111 = 可选择作为通道 DMAREQ 的 DMAIRQ0-DMAIRQ127

注 1： FORCE 位不能被用户清零。当强制的 DMA 传输完成时， FORCE 位由硬件清零。
2： 请参见表 6-1 获取所有中断源的 IRQ 编号的完整列表。

© 2007 Microchip Technology Inc.

寄存器 7-3： DMAxSTA：DMA 通道 x RAM 起始地址寄存器 A

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STA<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STA<7:0>							
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0 **STA<15:0>**：主 DMA RAM 起始地址位 （源地址或目标地址）

寄存器 7-4： DMAxSTB：DMA 通道 x RAM 起始地址寄存器 B

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STB<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STB<7:0>							
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0 **STB<15:0>**：辅助 DMA RAM 起始地址位 （源地址或目标地址）

寄存器 7-5：　DMAxPAD：DMA 通道 x 外设地址寄存器 (1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PAD<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PAD<7:0>							
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0　**PAD<15:0>：**外设地址寄存器位

注　1： 如果使能了通道 （即通道处于工作状态），写入该寄存器可能导致 DMA 通道的行为不可预测，应该避免。

寄存器 7-6：　DMAxCNT：DMA 通道 x 传输计数寄存器 (1)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	CNT<9:8>(2)	
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNT<7:0>(2)							
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-10　**未实现：**读为 0

bit 9-0　**CNT<9:0>：**DMA 传输计数寄存器位 (2)

注　1： 如果使能了通道 （即通道处于工作状态），写入该寄存器可能导致 DMA 通道的行为不可预测，应该避免。

2： DMA 传输的次数 = CNT<9:0> + 1。

© 2007 Microchip Technology Inc.

寄存器 7-7： DMACS0：DMA 控制器状态寄存器 0

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
PWCOL7	PWCOL6	PWCOL5	PWCOL4	PWCOL3	PWCOL2	PWCOL1	PWCOL0
bit 15							bit 8

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
XWCOL7	XWCOL6	XWCOL5	XWCOL4	XWCOL3	XWCOL2	XWCOL1	XWCOL0
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **PWCOL7：**通道 7 外设写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

bit 14 **PWCOL6：**通道 6 外设写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

bit 13 **PWCOL5：**通道 5 外设写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

bit 12 **PWCOL4：**通道 4 外设写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

bit 11 **PWCOL3：**通道 3 外设写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

bit 10 **PWCOL2：**通道 2 外设写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

bit 9 **PWCOL1：**通道 1 外设写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

bit 8 **PWCOL0：**通道 0 外设写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

bit 7 **XWCOL7：**通道 7 DMA RAM 写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

bit 6 **XWCOL6：**通道 6 DMA RAM 写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

bit 5 **XWCOL5：**通道 5 DMA RAM 写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

bit 4 **XWCOL4：**通道 4 DMA RAM 写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

寄存器 7-7：　　DMACS0：DMA 控制器状态寄存器 0 （续）

bit 3　　**XWCOL3**：通道 3 DMA RAM 写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

bit 2　　**XWCOL2**：通道 2 DMA RAM 写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

bit 1　　**XWCOL1**：通道 1 DMA RAM 写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

bit 0　　**XWCOL0**：通道 0 DMA RAM 写冲突标志位
1 = 检测到写冲突
0 = 未检测到写冲突

© 2007 Microchip Technology Inc.

寄存器 7-8：　　DMACS1：DMA 控制器状态寄存器 1

U-0	U-0	U-0	U-0	R-1	R-1	R-1	R-1
—	—	—	—	LSTCH<3:0>			
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
PPST7	PPST6	PPST5	PPST4	PPST3	PPST2	PPST1	PPST0
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-12　**未实现：**读为 0

bit 11-8　**LSTCH<3:0>：**上一次工作的 DMA 通道位
1111 = 自系统复位以来没有发生 DMA 传输
1110-1000 = 保留
0111 = 上次数据传输是通过 DMA 通道 7 进行的
0110 = 上次数据传输是通过 DMA 通道 6 进行的
0101 = 上次数据传输是通过 DMA 通道 5 进行的
0100 = 上次数据传输是通过 DMA 通道 4 进行的
0011 = 上次数据传输是通过 DMA 通道 3 进行的
0010 = 上次数据传输是通过 DMA 通道 2 进行的
0001 = 上次数据传输是通过 DMA 通道 1 进行的
0000 = 上次数据传输是通过 DMA 通道 0 进行的

bit 7　**PPST7：**通道 7 乒乓模式状态标志位
1 = 选择 DMA7STB 寄存器
0 = 选择 DMA7STA 寄存器

bit 6　**PPST6：**通道 6 乒乓模式状态标志位
1 = 选择 DMA6STB 寄存器
0 = 选择 DMA6STA 寄存器

bit 5　**PPST5：**通道 5 乒乓模式状态标志位
1 = 选择 DMA5STB 寄存器
0 = 选择 DMA5STA 寄存器

bit 4　**PPST4：**通道 4 乒乓模式状态标志位
1 = 选择 DMA4STB 寄存器
0 = 选择 DMA4STA 寄存器

bit 3　**PPST3：**通道 3 乒乓模式状态标志位
1 = 选择 DMA3STB 寄存器
0 = 选择 DMA3STA 寄存器

bit 2　**PPST2：**通道 2 乒乓模式状态标志位
1 = 选择 DMA2STB 寄存器
0 = 选择 DMA2STA 寄存器

bit 1　**PPST1：**通道 1 乒乓模式状态标志位
1 = 选择 DMA1STB 寄存器
0 = 选择 DMA1STA 寄存器

bit 0　**PPST0：**通道 0 乒乓模式状态标志位
1 = 选择 DMA0STB 寄存器
0 = 选择 DMA0STA 寄存器

寄存器 7-9：　　DSADR：最近的 DMA RAM 地址

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
DSADR<15:8>							
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
DSADR<7:0>							
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0　　**DSADR<15:0>：**DMA 控制器最近访问的 DMA RAM 地址位

© 2007 Microchip Technology Inc.

8.0 振荡器配置

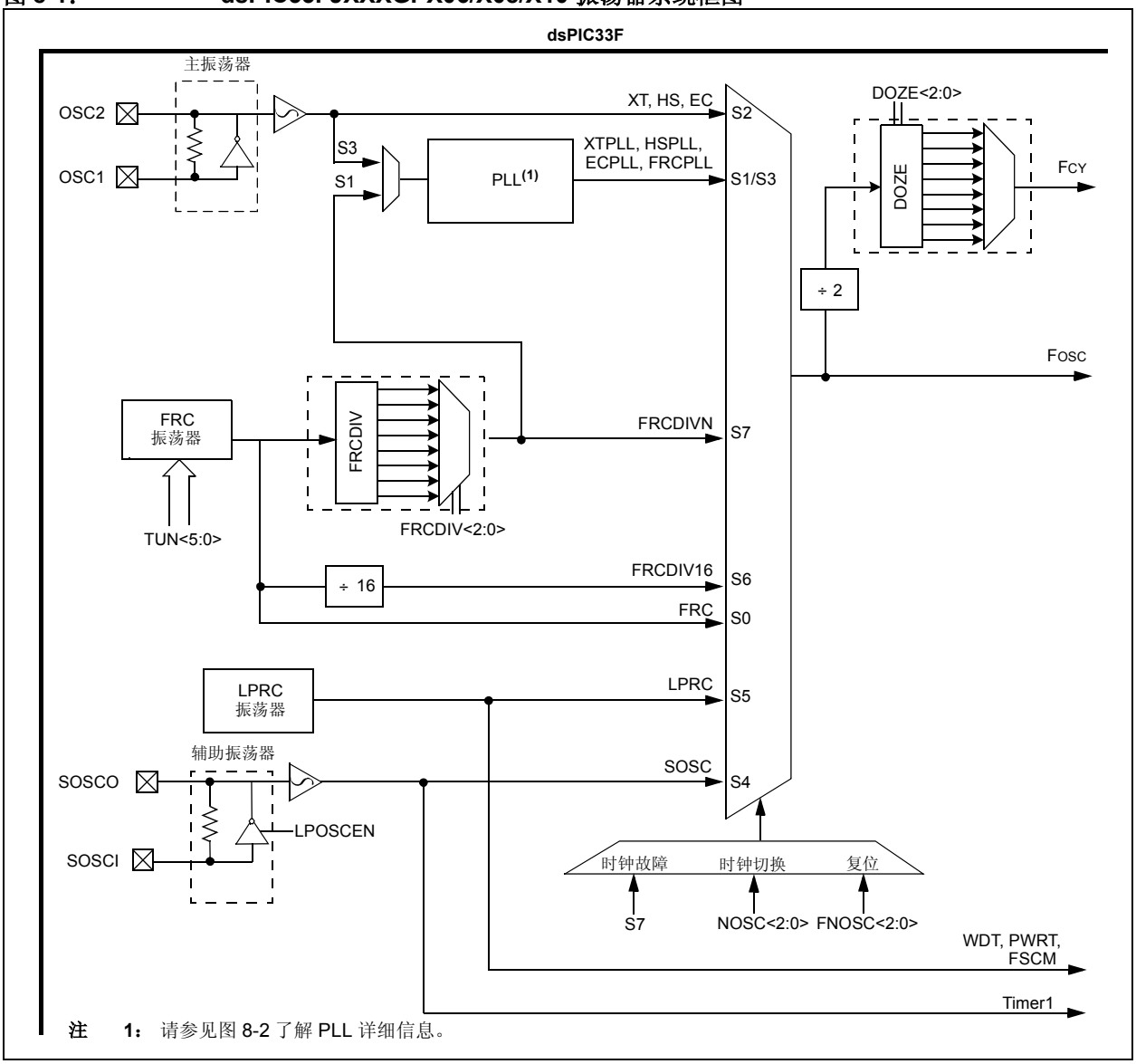
注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10 器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

dsPIC33FJXXXGPX06/X08/X10 振荡器系统提供：

- 可选择多种外部和内部振荡器作为时钟源
- 片上PLL可将内部工作频率调整为所要求系统时钟频率
- 内部FRC振荡器也可使用PLL，因此允许在没有任何外部时钟产生硬件的情况下全速工作
- 不同时钟源之间的时钟切换
- 可节省系统功耗的可编程时钟后分频器
- 故障保护时钟监视器（FSCM），可检测时钟故障并采取故障保护措施
- 一个时钟控制寄存器（OSCCON）
- 用于主振荡器选择的非易失性配置位

图 8-1 给出了振荡器系统的简化框图。

图 8-1： dsPIC33FJXXXGPX06/X08/X10 振荡器系统框图



注 1： 请参见图 8-2 了解 PLL 详细信息。

8.1 CPU 时钟系统

dsPIC33FJXXXGPX06/X08/X10 器件提供 7 种系统时钟选择：

- FRC 振荡器
- 带 PLL 的 FRC 振荡器
- 主 （XT、 HS 或 EC）振荡器
- 带 PLL 的主振荡器
- 辅助 （LP）振荡器
- LPRC 振荡器
- 带后分频器的 FRC 振荡器

8.1.1 系统时钟源

FRC（快速RC）内部振荡器工作频率的标称值为7.37 MHz。用户软件可以调节 FRC 频率。用户软件能够通过有选择地指定 FRC 时钟的分频比（从 1:2 至 1:256）。使用 FRCDIV<2:0> （CLKDIV<10:8>）位来选择该分频比。

主振荡器能以下列任一时钟作为其时钟源：

1. XT（晶振）：3 MHz 至 10 MHz 范围内的晶振和陶瓷谐振器。晶振连接在 OSC1 和 OSC2 引脚之间。
2. HS （高速晶振）：10 MHz 至 40 MHz 范围内的晶振。晶振连接在 OSC1 和 OSC2 引脚之间。
3. EC（外部时钟）：0.8 MHz 至 64 MHz 范围内的外部时钟信号。外部时钟信号直接施加到 OSC1 引脚。

辅助 （LP）振荡器是为低功耗运行而设计的，它使用 32.768 kHz 晶振或陶瓷谐振器。LP 振荡器使用 SOSCI 和 SOSCO 引脚。

LPRC （低功耗 RC）内部振荡器工作频率的标称值为 32.768 kHz。它也可以用作看门狗定时器 （WDT）和故障保护时钟监视器 （FSCM）的参考时钟。

可选择将 FRC 和主振荡器产生的时钟信号加到片上锁相环（PLL），为器件工作提供宽范围的输出频率。PLL 配置如**第 8.1.3 节 “PLL 配置”**中所述。

8.1.2 系统时钟选择

通过设置配置位可选择器件发生上电复位事件时使用的振荡器源。振荡器配置位设置位于程序存储器的配置寄存器中。（更多详细信息，请参见**第 21.1 节 “配置位”** 。）初始振荡器选择配置位 FNOSC<2:0> （FOSCSEL<2:0>）和主振荡器模式选择配置位 POSCMD<1:0> （FOSC<1:0>）选择在上电复位时使用的振荡器源。 FRC 主振荡器是默认的 （未编程的）选择。

配置位允许用户在12种不同的时钟模式之间进行选择，如表 8-1 所示。

振荡器的输出（或当选择了 PLL 模式时 PLL 的输出）FOSC 被 2 分频以产生器件指令时钟 （FCY）。 FCY 定义器件的工作速度，dsPIC33FJXXXGPX06/X08/X10 架构可支持最高 40 MHz 的工作速度。

指令执行速度或器件工作频率 FCY 如下：

公式 8-1： 器件工作频率

$$F_{CY} = F_{OSC}/2$$

8.1.3 PLL 配置

主振荡器和内部 FRC 振荡器能有选择地使用片上 PLL 来获取更高的工作速度。PLL 在选择器件工作速度方面提供很大的灵活性。图 8-2 给出了 PLL 的框图。

以 FIN 表示的主振荡器或 FRC 的输出在提供给 PLL 的压控振荡器（Voltage Controlled Oscillator， VCO）之前被预分频因子（N1）2、3、... 或 33 分频。VCO 的输入必须在 0.8 MHz 到 8 MHz 的范围内进行选择。由于最小的预分频因子为 2，这意味着必须在 1.6 MHz 到 16 MHz 的范围内选择 FIN。使用 PLLPRE<4:0> 位（CLKDIV<4:0>）来选择预分频因子 “N1”。

可由 PLLDIV<8:0> 位（PLLFBD<8:0>）选择 PLL 反馈倍频比，这些位提供可使VCO的输入倍频的因子“M”。必须选择该因子以使产生的 VCO 输出频率在 100 MHz 到 200 MHz 范围内。

VCO 输出进一步被后分频因子 “N2”分频。使用 PLLPOST<1:0> 位 （CLKDIV<7:6>）来选择该因子。“N2”可以是 2、 4 或 8，必须选择该因子，以使 PLL 输出频率（FOSC）在 12.5 MHz 到 80 MHz 范围内，以产生 6.25-40 MIPS 的器件工作速度。

对于主振荡器或 FRC 振荡器，输出为 FIN，则 PLL 的输出 FOSC 为 （由以下公式计算）：

公式 8-2： FOSC 计算

$$F_{OSC} = F_{IN} * \left(\frac{M}{N1 * N2} \right)$$

© 2007 Microchip Technology Inc.

例如，假设正在使用 10 MHz 晶振，“带 PLL 的 XT”为所选的振荡器模式。如果 PLLPRE<4:0> = 0，那么 N1 = 2。这将产生频率为 10/2 = 5 MHz 的 VCO 输入，这一频率在 0.8-8 MHz 的可接受范围内。如果 PLLDIV<8:0> = 0x1E，那么 M = 32。这将产生频率为 5 x 32 = 160 MHz 的 VCO 输出，这一频率在所需的 100-200 MHz 范围内。

如果 PLLPOST<1:0> = 0，那么 N2 = 2。这提供 160/2 = 80 MHz 的 Fosc。产生的器件工作速度是 80/2 = 40 MIPS。

公式 8-3：　　带 PLL 的 XT 模式示例

$$FCY = \frac{FOSC}{2} = \frac{1}{2}\left(\frac{10000000*32}{2*2}\right) = 40 \text{ MIPS}$$

图 8-2：　　dsPIC33FJXXXGPX06/X08/X10 PLL 框图

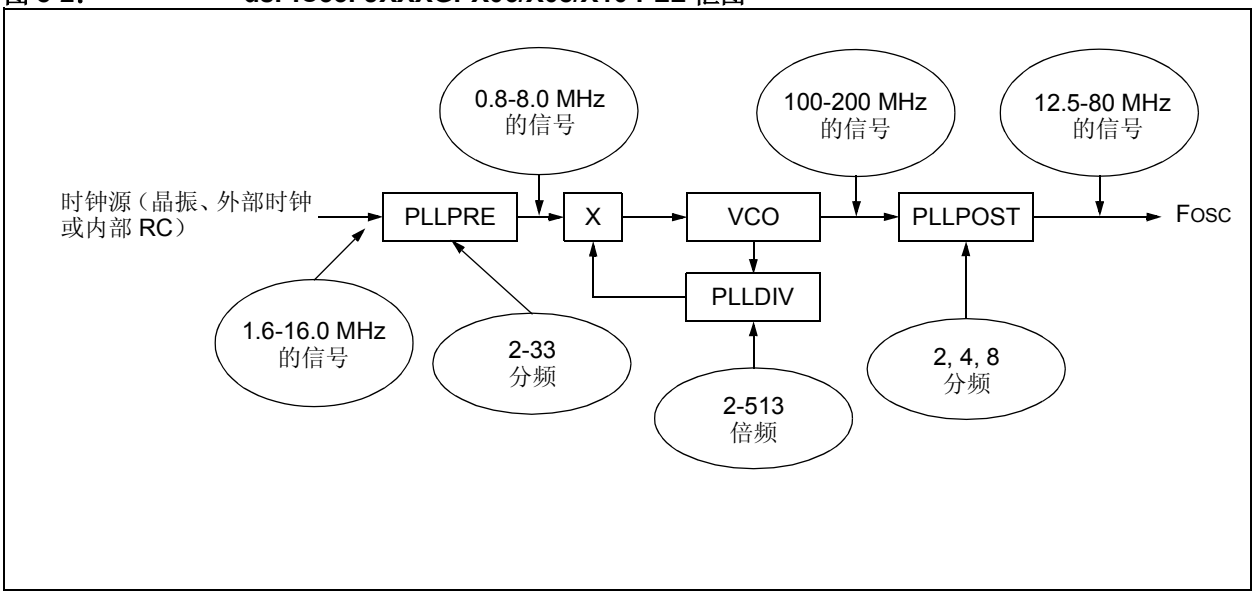


表 8-1：　　时钟选择的配置位值

振荡器模式	振荡器源	POSCMD<1:0>	FNOSC<2:0>	注
带 N 分频的快速 RC 振荡器（FRCDIVN）	内部	xx	111	**1, 2**
带 16 分频的快速 RC 振荡器（FRCDIV16）	内部	xx	110	**1**
低功耗 RC 振荡器 （LPRC）	内部	xx	101	**1**
辅助 （Timer1）振荡器 （SOSC）	辅助	xx	100	**1**
带 PLL 的主振荡器 （HS）（HSPLL）	主	10	011	
带 PLL 的主振荡器 （XT）（XTPLL）	主	01	011	
带 PLL 的主振荡器 （EC）（ECPLL）	主	00	011	**1**
主振荡器 （HS）	主	10	010	
主振荡器 （XT）	主	01	010	
主振荡器 （EC）	主	00	010	**1**
带 PLL 的快速 RC 振荡器 （FRCPLL）	内部	xx	001	**1**
快速 RC 振荡器 （FRC）	内部	xx	000	**1**

注　1： OSC2 引脚功能由 OSCIOFNC 配置位决定。

2： 对于未编程 （已擦除）器件，这是默认的振荡器模式。

寄存器 8-1： OSCCON：振荡器控制寄存器

U-0	R-0	R-0	R-0	U-0	R/W-y	R/W-y	R/W-y
—	COSC<2:0>			—	NOSC<2:0>		
bit 15							bit 8

R/W-0	U-0	R-0	U-0	R/C-0	U-0	R/W-0	R/W-0
CLKLOCK	—	LOCK	—	CF	—	LPOSCEN	OSWEN
bit 7							bit 0

图注：	y = 在 POR 时由配置位设置的值		
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **未实现：**读为 0

bit 14-12 **COSC<2:0>：**当前振荡器选择位 （只读）
000 = 快速 RC 振荡器 （FRC）
001 = 带 PLL 的快速 RC 振荡器 （FRC）
010 = 主振荡器 （XT、 HS 和 EC）
011 = 带 PLL 的主振荡器 （XT、 HS 和 EC）
100 = 辅助振荡器 （SOSC）
101 = 低功耗 RC 振荡器 （LPRC）
110 = 带 16 分频的快速 RC 振荡器 （FRC）
111 = 带 n 分频的快速 RC 振荡器 （FRC）

bit 11 **未实现：**读为 0

bit 10-8 **NOSC<2:0>：**新振荡器选择位
000 = 快速 RC 振荡器 （FRC）
001 = 带 PLL 的快速 RC 振荡器 （FRC）
010 = 主振荡器 （XT、 HS 和 EC）
011 = 带 PLL 的主振荡器 （XT、 HS 和 EC）
100 = 辅助振荡器 （SOSC）
101 = 低功耗 RC 振荡器 （LPRC）
110 = 带 16 分频的快速 RC 振荡器 （FRC）
111 = 带 n 分频的快速 RC 振荡器 （FRC）

bit 7 **CLKLOCK：**时钟锁定使能位
1 = 如果 （FCKSM0 = 1），那么时钟和 PLL 配置被锁定。
如果 （FCKSM0 = 0），那么时钟和 PLL 配置可以被修改。
0 = 时钟和 PLL 选择未被锁定，配置可以被修改

bit 6 **未实现：**读为 0

bit 5 **LOCK：**PLL 锁定状态位 （只读）
1 = 表示 PLL 处于锁定状态，或 PLL 起振定时器延时结束
0 = 表示 PLL 处于失锁状态，起振定时器在运行或 PLL 被禁止

bit 4 **未实现：**读为 0

bit 3 **CF：**时钟故障检测位 （由应用程序读 / 清零）
1 = FSCM 检测到时钟故障
0 = FSCM 未检测到时钟故障

bit 2 **未实现：**读为 0

bit 1 **LPOSCEN：**辅助 （LP）振荡器使能位
1 = 使能辅助振荡器
0 = 禁止辅助振荡器

bit 0 **OSWEN：**振荡器切换使能位
1 = 请求切换到由 NOSC<2:0> 位指定的振荡器
0 = 振荡器切换完成

© 2007 Microchip Technology Inc.

寄存器 8-2：　　CLKDIV：时钟分频比寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0
ROI	DOZE<2:0>			DOZEN(1)	FRCDIV<2:0>		
bit 15							bit 8

R/W-0	R/W-1	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PLLPOST<1:0>		—	PLLPRE<4:0>				
bit 7							bit 0

图注：	y = 在 POR 时由配置位设置的值		
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**ROI：**中断恢复位
1 = 中断将清零 DOZEN 位，并且处理器时钟 / 外设时钟比被设置为 1:1
0 = 中断对 DOZEN 位无影响

bit 14-12　**DOZE<2:0>：**处理器时钟分频比选择位
000 = FCY/1
001 = FCY/2
010 = FCY/4
011 = FCY/8 （默认）
100 = FCY/16
101 = FCY/32
110 = FCY/64
111 = FCY/128

bit 11　**DOZEN：**DOZE 模式使能位 (1)
1 = DOZE<2:0> 位域指定外设时钟和处理器时钟之间的比率
0 = 处理器时钟 / 外设时钟比率强制为 1:1

bit 10-8　**FRCDIV<2:0>：**内部快速 RC 振荡器后分频比位
000 = FRC 1 分频 （默认）
001 = FRC 2 分频
010 = FRC 4 分频
011 = FRC 8 分频
100 = FRC 16 分频
101 = FRC 32 分频
110 = FRC 64 分频
111 = FRC 256 分频

bit 7-6　**PLLPOST<1:0>：**PLL VCO 输出分频比选择位 （也表示为 “N2”， PLL 后分频比）
00 = 输出 /2
01 = 输出 /4 （默认）
10 = 保留
11 = 输出 /8

bit 5　**未实现：**读为 0

bit 4-0　**PLLPRE<4:0>：**PLL 相位检测器输入分频比位 （也表示为 “N1”， PLL 预分频比）
00000 = 输入 /2 （默认）
00001 = 输入 /3
•
•
•
11111 = 输入 /33

注　1： 该位在 ROI 位置 1 和发生中断时清零。

寄存器 8-3：　　PLLFBD：PLL 反馈倍频比寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0(1)
—	—	—	—	—	—	—	PLLDIV<8>
bit 15							bit 8

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
PLLDIV<7:0>							
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-9　**未实现：**读为 0

bit 8-0　**PLLDIV<8:0>：**PLL 反馈倍频比位（也表示为 “M”， PLL 倍频比）

000000000 = 2
000000001 = 3
000000010 = 4
•
•
•
000110000 = 50（默认）
•
•
•
111111111 = 513

© 2007 Microchip Technology Inc.

寄存器 8-4：　　OSCTUN：FRC 振荡器调节寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-6　**未实现：**读为 0

bit 5-0　**TUN<5:0>：**FRC 振荡器调节位

011111 = 中心频率 + 11.625%
011110 = 中心频率 + 11.25% （8.23 MHz）
•
•
•
000001 = 中心频率 + 0.375% （7.40 MHz）
000000 = 中心频率 （标称值 7.37 MHz）
111111 = 中心频率 – 0.375% （7.345 MHz）
•
•
•
100001 = 中心频率 – 11.625% （6.52 MHz）
100000 = 中心频率 – 12% （6.49 MHz）

8.2 时钟切换工作原理

在软件控制下，应用可以在任何时候在四个时钟源（主振荡器、LP、FRC 和 LPRC）之间自由切换。为限制这种灵活性可能产生的影响， dsPIC33FJXXXGPX06/X08/X10 器件的时钟切换过程带有安全锁定。

注：	主振荡器模式有三种不同的子模式 （XT、HS 和 EC），它们由 POSCMD<1:0> 配置位决定。在应用中可以用软件实现从主振荡器模式切换到其他模式，或从其他模式切换到主振荡器模式，但不能在不对器件进行再编程的情况下在主振荡器模式的不同子模式之间进行切换。

8.2.1 使能时钟切换

要使能时钟切换，配置寄存器中的 FCKSM1 配置位必须编程为 0。（更多详细信息，请参见**第 21.1 节“配置位”**。）如果 FCKSM1 配置位未被编程 （为 1），则时钟切换功能和故障保护监视器功能被禁止。这是默认设置。

当时钟切换被禁止时，NOSC 控制位（OSCCON<10:8>）不控制时钟选择。但是，COSC 位（OSCCON<14:12>）反映由 FNOSC 配置位选择的时钟源。

在时钟切换被禁止时，OSWEN 控制位（OSCCON<0>）无效。它总是保持为 0。

8.2.2 振荡器切换过程

执行时钟切换至少需要以下基本过程：

1. 根据需要读 COSC 位 （OSCCON<14:12>）以确定当前的振荡器源。
2. 执行解锁序列以允许写入 OSCCON 寄存器的高字节。
3. 将适当的值写入新振荡器源的 NOSC 控制位（OSCCON<10:8>）。
4. 执行解锁序列以允许写入 OSCCON 寄存器的低字节。
5. 将 OSWEN 位置 1 以启动振荡器切换。

一旦基本过程完成，系统时钟硬件将自动进行如下响应：

1. 时钟切换硬件将 NOSC 控制位的新值与 COSC 状态位进行比较。如果它们相同，则时钟切换是多余的操作。在这种情况下，OSWEN 位自动清零，时钟切换中止。
2. 如果启动了有效的时钟切换，则 LOCK （OSCCON<5>）和 CF （OSCCON<3>）状态位清零。
3. 如果新振荡器现在不在运行，则硬件会将它开启。如果开启的是晶振，则硬件将等待直到振荡器起振定时器（OST）超时。如果新的振荡器源使用 PLL，则硬件将等待直到检测到 PLL 锁定（LOCK = 1）。
4. 硬件会等待新时钟源的 10 个时钟周期，然后执行时钟切换。
5. 硬件清零 OSWEN 位表示时钟转换成功。此外，NOSC 位的值被传送到 COSC 状态位。
6. 此时旧时钟源被关闭， LPRC （如果 WDT 或 FSCM 被使能）或 LP （如果 LPOSCEN 保持置 1）除外。

注	**1：** 在整个时钟切换过程中，处理器将继续执行代码。对时序敏感的代码不应在此时执行。
	2： 不允许直接在使能 PLL 的任何主振荡器模式和 FRCPLL 之间进行时钟切换。这适用于任何方向下的时钟切换。在这些情况下，应用必须首先切换到 FRC 模式将其作为两个 PLL 模式之间的过渡时钟源。

8.3 故障保护时钟监视器 （FSCM）

故障保护时钟监视器 （FSCM）允许器件在即使振荡器发生故障的情况下仍能继续运行。通过编程使能 FSCM 功能。如果使能了 FSCM 功能，LPRC 内部振荡器将总是运行 （休眠模式除外），并且不受看门狗定时器的控制。

在发生振荡器故障时， FSCM 会产生时钟故障陷阱事件，并将系统时钟切换到 FRC 振荡器。然后应用程序可尝试重新启动振荡器或执行受控关闭。通过简单地将复位地址装入振荡器故障陷阱向量，可将陷阱作为一个热复位来处理。

如果使用 PLL 倍频器来对系统时钟倍频，则时钟发生故障时内部 FRC 也会被以相同的倍频比倍频。时钟发生故障时器件会切换到带 PLL 的 FRC。

© 2007 Microchip Technology Inc.

9.0 节能特性

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

dsPIC33FJXXXGPX06/X08/X10 器件提供了管理功耗的功能，该功能是通过有选择地管理 CPU 和外设的时钟源来实现的。一般而言，较低的时钟频率和减少时钟源驱动电路的数目会使功耗降低。dsPIC33FJXXXGPX06/X08/X10 器件用以下四种方法管理功耗：

- 时钟频率
- 基于指令的休眠模式和空闲模式
- 软件控制的打盹模式
- 用软件有选择地进行外设控制

可以组合使用这些方法从而在保证关键应用特性（如对于时序敏感的通信）的情况下有选择地调节应用的功耗。

9.1 时钟频率和时钟切换

dsPIC33FJXXXGPX06/X08/X10 器件提供的时钟频率范围较大，用户可根据应用需要进行选择。如果未锁定系统时钟配置，用户只需更改 NOSC 位（OSCCON<10:8>）即可选择低功耗或高精度振荡器。在工作期间更改系统时钟的过程以及相应的限制，将在**第 8.0 节 “振荡器配置”**中进行更详细的讨论。

9.2 基于指令的节能模式

dsPIC33FJXXXGPX06/X08/X10 器件有两种特殊的节能模式，通过执行特殊的 PWRSAV 指令可以进入这两种模式。休眠模式下时钟停止工作并停止所有代码执行；空闲模式下 CPU 停止工作并停止代码执行，但是允许外设模块继续工作。例 9-1 中所示为 PWRSAV 指令的汇编语法。

注： SLEEP_MODE 和 IDLE_MODE 是在所选器件的汇编头文件中定义的常量。

在被允许的中断产生、WDT 超时或器件复位时，器件会退出休眠和空闲模式。器件退出这两种模式的过程称为“唤醒”。

9.2.1 休眠模式

休眠模式具有下列特征：

- 系统时钟源关闭。如果使用了片上振荡器，也要关闭它。
- 如果 I/O 引脚上不消耗电流，则器件电流消耗将降至最低。
- 由于系统时钟源被禁止，所以故障保护时钟监视器在休眠模式下不工作。
- 如果WDT被使能，则LPRC时钟将继续在休眠模式下运行。
- 如果 WDT 被使能，则在进入休眠模式之前自动清零。
- 有些器件功能或外设可能在休眠模式下继续工作，包括 I/O 端口上的输入电平变化通知功能或使用外部时钟输入的外设等。任何需要使用系统时钟源来工作的外设在休眠模式下将被禁止。

当发生以下任何事件时，器件将从休眠模式唤醒：

- 产生任何被单独允许的中断。
- 任何形式的器件复位。
- WDT 超时。

从休眠模式唤醒时，处理器将使用在进入休眠模式时处于工作状态的时钟源重新开始工作。

例 9-1： PWRSAV 指令语法

```
PWRSAV  #SLEEP_MODE    ; Put the device into SLEEP mode
PWRSAV  #IDLE_MODE     ; Put the device into IDLE mode
```

9.2.2 空闲模式

空闲模式具有下列特征：

- CPU 将停止执行指令。
- WDT 被自动清零。
- 系统时钟源保持工作状态。默认情况下，所有外设模块将继续使用系统时钟源正常工作，也可以有选择地禁止它们（见**第 9.4 节 “外设模块禁止”**）。
- 如果WDT或FSCM被使能，则LPRC也将保持工作状态。

当发生以下任何事件时，器件将从空闲模式唤醒：

- 产生任何被单独允许的中断。
- 任何器件复位。
- WDT 超时。

从空闲模式唤醒时，重新为 CPU 提供时钟，且立即从 PWRSAV 指令之后的下一条指令或 ISR 中的第一条指令开始执行指令。

9.2.3 在节能指令执行期间的中断

在 PWRSAV 指令执行期间发生的中断都将延迟到进入休眠或空闲模式后才产生，并导致器件从休眠或空闲模式中唤醒。

9.3 打盹模式

通常，更改时钟速度和进入某种节能模式是降低功耗的首选策略。然而，有些情况下不可行。例如，某些应用可能必须保持不间断的同步通信，即便在它不执行任何其他操作时也不例外。降低系统时钟速度可能会带来通信错误，而使用节能模式可能会完全终止通信。

打盹模式是另一种简单有效的节能方法，它可以在器件仍然执行代码的情况下降低功耗。在此模式下，系统时钟以相同的时钟源和相同的速度继续工作。外设模块时钟速度保持不变，但 CPU 时钟的速度降低了。保持这两个时钟域同步，可以保持外设访问 SFR 的能力，同时 CPU 以较慢的速度执行代码。

通过将 DOZEN 位（CLKDIV<11>）置 1 使能打盹模式。外设与内核的时钟速度之比是由 DOZE<2:0> 位（CLKDIV<14:12>）决定的。有八种可能的配置，从 1:1 到 1:128，其中 1:1 是默认设置。

在事件驱动的应用中，使用打盹模式有选择地降低功耗是可行的。这样就可以实现不间断地运行对时序要求高的功能（如同步通信），而 CPU 保持空闲等待事件调用中断服务程序。通过将 ROI 位（CLKDIV<15>）置 1，可以使器件在产生中断时自动返回到全速 CPU 工作模式。默认情况下，中断事件对打盹模式工作没有影响。

例如，假设器件的工作速度为 20 MIPS，并根据这一速度将 CAN 模块的速度配置为 500 kbps。如果现在将器件置于时钟频率比为 1:4 的打盹模式下，那么 CAN 模块将继续按要求的 500 kbps 位速率通信，而 CPU 则以 5 MIPS 的速度开始执行指令。

9.4 外设模块禁止

外设模块禁止（Peripheral Module Disable，PMD）寄存器通过停止所有提供给模块的时钟源提供一种禁止外设模块的方法。当通过相应的 PMD 控制位禁止外设时，外设就进入了功耗最低的状态。与外设相关的控制寄存器和状态寄存器也被禁止，因此写入那些寄存器不会有影响，且读取值无效。

只有在 PMD 寄存器中的相应位被清零且特定的 dsPIC® DSC 器件支持某个外设时，才会使能相应的外设模块。如果外设包含在器件中，则默认情况下，它在 PMD 寄存器中是使能的。

注：	如果 PMD 位置 1，则对应的模块将在一个指令周期的延时后被禁止。类似地，如果 PMD 位清零，则对应的模块将在一个指令周期的延时后被使能（假设已将模块控制寄存器配置为使能模块的工作）。

© 2007 Microchip Technology Inc.

10.0 I/O 端口

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

所有器件引脚（VDD、VSS、MCLR 和 OSC1/CLKIN 除外）均由外设和并行 I/O 端口所共用。所有 I/O 输入端口都为施密特触发器输入，提高了抗噪声能力。

10.1 并行 I/O （PIO）端口

通常，与某个外设共用一个引脚的并行 I/O 端口总是服从于该外设。外设的输出缓冲器数据和控制信号提供给一对多路开关。这对多路开关用于选择 I/O 引脚的输出数据和控制信号是用于外设还是相应的端口。该逻辑电路同时会阻止“环回进入”（loop through），即一个端口的数字输出可以驱动共用同一个引脚的外设输入。图 10-1 显示了端口是如何与其他外设复用的以及相关的 I/O 引脚。

如果外设使能，并且正在驱动相关引脚时，该引脚将不再作为通用 I/O 引脚使用。可以读该 I/O 引脚，但并行端口位的输出驱动器将被禁止。如果使能某外设但没有驱动引脚时，则该引脚可由端口驱动。

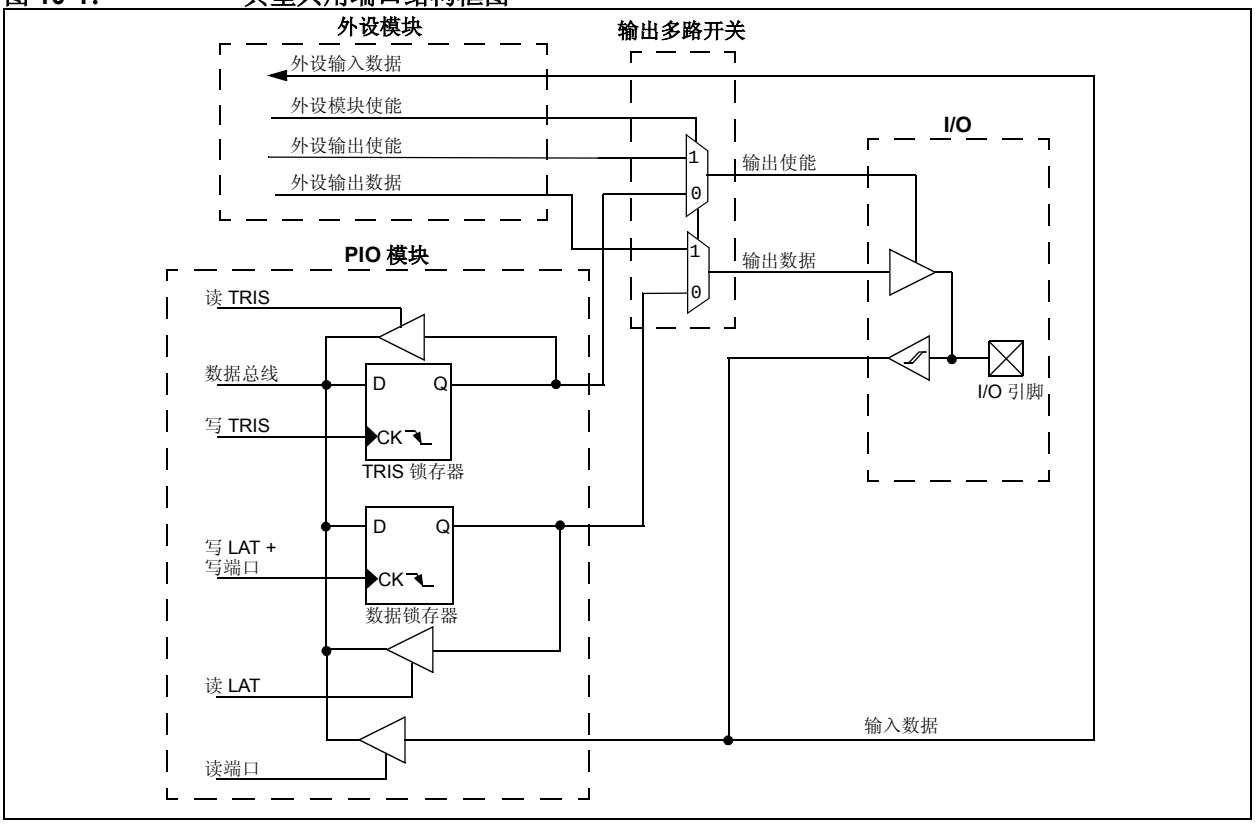
所有端口引脚都有3个寄存器与端口引脚作为数字I/O时的工作直接相关。数据方向寄存器（TRISx）决定引脚是输入还是输出。如果数据方向位为 1，则引脚是输入。复位后，所有端口引脚均定义为输入。读锁存器（LATx）时，读到的是锁存器中的值；写锁存器时，写入的是锁存器。读端口（PORTx）时，读到的是端口引脚的值；而写端口引脚时，写入的是锁存器。

对于特定器件无效的任何位及其相关的数据和控制寄存器都将被禁止。这意味着对应的 LATx 和 TRISx 寄存器以及端口引脚都将读为零。

当端口引脚与另一个外设共用或与定义为仅输入的功能共用时，它将被视为专用端口，因为没有任何其他竞争的输出源。 INT4 引脚就是这样一个例子。

注： 数字输入引脚上的电压可在 -0.3V 至 5.6V 之间。

图 10-1： 典型共用端口结构框图



10.2 漏极开路配置

除 PORT、 LAT 和 TRIS 寄存器用于数据控制外，每个端口引脚也可被单独地配置为数字输出或漏极开路输出。这是由与每个端口相关的漏极开路控制寄存器 ODCx 控制的，将其中的任何位置 1 即可将相应的引脚配置为漏极开路输出。

这种漏极开路特性允许通过使用外部上拉电阻，在所需的任意仅用作数字功能的引脚上产生高于 VDD（如 5V）的输出电平。（与模拟功能复用的引脚不支持漏极开路 I/O 特性。）允许的最大开漏电压与最大 VIH 规范相同。端口引脚和外设配置都支持漏极开路输出特性。

10.3 配置模拟端口引脚

ADxPCFGH、ADxPCFGL和TRIS寄存器用于控制ADC端口引脚的操作。如果要将端口引脚用作模拟输入，则对应的 TRIS 位必须置 1（输入）。如果将 TRIS 位清零（输出），则数字输出电平 （VOH 或 VOL）将被转换。

清零 ADxPCFGH 或 ADxPCFGL 寄存器中的任何位都会将相应的引脚配置为模拟引脚。这也是与模拟（ANx）功能相关的任何 I/O 引脚的复位状态。

注：	在有两个 ADC 模块的器件中，如果在 AD1PCFGH（L）和 AD2PCFGH（L）中相应的 PCFG 位被清零，则相应的引脚将被配置为模拟输入。

当读取端口寄存器时，所有配置为模拟输入通道的引脚均读为零 （低电平）。

配置为数字输入的引脚将不会对模拟输入进行转换。对于任何定义为数字输入的引脚（包括 ANx 引脚），加在引脚上的模拟电压可能导致输入缓冲器消耗的电流超出器件规范。

注：	模拟输入引脚上的电压可以在 -0.3V 至（VDD + 0.3 V）之间。

10.4 I/O 端口写 / 读时序

端口方向改变或端口写操作与同一端口的读操作之间需要一个指令周期。通常该指令是 NOP。

10.5 输入状态变化通知

I/O 端口的输入状态变化通知功能允许 dsPIC33FJXXXGPX06/X08/X10 器件在选定输入引脚的状态变化时，向处理器发出中断请求。当禁止时钟时，该特性还可在休眠模式下检测到输入状态变化。根据器件的引脚数，最多可以选择 （允许） 24 个外部信号（CN0 到 CN23）在输入状态发生变化时产生中断请求。

有 4 个与 CN 模块相关的控制寄存器。CNEN1 和 CNEN2 寄存器包含每个 CN 输入引脚的 CN 中断允许（CNxIE）控制位。将其中任一位置 1 将允许相应引脚的 CN 中断。

每个 CN 引脚都有一个与之相连的弱上拉电路。弱上拉电路充当连接到该引脚的电流源，当连接了按钮或键盘设备时，不再需要使用外部电阻。使用包含每个 CN 引脚弱上拉使能 （CNxPUE）位的 CNPU1 和 CNPU2 寄存器可分别使能各个上拉电路。将任一控制位置 1 均可使能相应引脚的弱上拉功能。

注：	当端口引脚被配置为数字输出时，状态变化通知引脚的上拉电路将始终被禁止。

例 10-1： 端口写 / 读示例

```
MOV   0xFF00, W0          ; Configure PORTB<15:8> as inputs
MOV   W0, TRISB           ; and PORTB<7:0> as outputs
NOP                       ; Delay 1 cycle
btss  PORTB, #13          ; Next Instruction
```


© 2007 Microchip Technology Inc.

11.0 TIMER1

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10 器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

Timer1 模块是一个 16 位定时器，可作为实时时钟的时间计数器，或作为自由运行的间隔定时器 / 计数器。Timer1 可在以下三种模式下工作：

- 16 位定时器
- 16 位同步计数器
- 16 位异步计数器

Timer1 还支持以下功能：

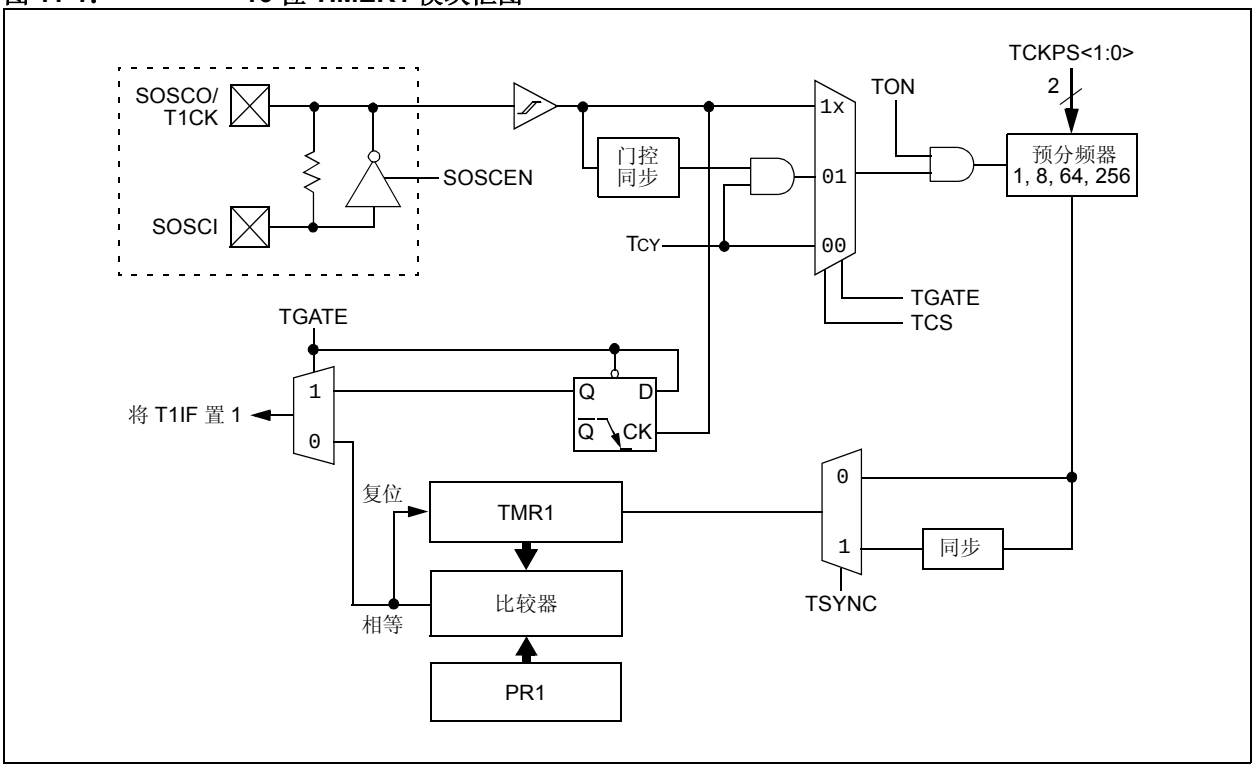
- 定时器门控操作
- 可选择的预分频比设置
- 在 CPU 空闲和休眠模式期间的定时器操作
- 在 16 位周期寄存器匹配时或外部门控信号的下降沿产生中断

图 11-1 给出了 16 位定时器模块的框图。

配置 Timer1 的操作：

1. 将 T1CON 寄存器中的 TON 位置 1 （= 1）。
2. 使用 T1CON 寄存器中的 TCKPS<1:0> 位选择定时器预分频比。
3. 使用 T1CON 寄存器中的 TCS 和 TGATE 位设置时钟和门控模式。
4. 将 T1CON 中的 TSYNC 位置 1 或清零来选择同步或异步操作。
5. 将定时器的周期值装入 PR1 寄存器。
6. 如果需要中断，将中断允许位 T1IE 置 1。使用优先级位 T1IP<2:0> 来设置中断优先级。

图 11-1： 16 位 TIMER1 模块框图



寄存器 11-1：　T1CON：TIMER1 控制寄存器

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON	—	TSIDL	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0
—	TGATE	TCKPS<1:0>		—	TSYNC	TCS	—
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**TON**：Timer1 使能位
1 = 启动 16 位 Timer1
0 = 停止 16 位 Timer1

bit 14　**未实现**：读为 0

bit 13　**TSIDL**：空闲模式停止位
1 = 当器件进入空闲模式时，模块停止工作
0 = 在空闲模式下模块继续工作

bit 12-7　**未实现**：读为 0

bit 6　**TGATE**：Timer1 门控时间累加使能位
当 T1CS = 1 时：
该位为无关位。
当 T1CS = 0 时：
1 = 使能门控时间累加
0 = 禁止门控时间累加

bit 5-4　**TCKPS<1:0>**：Timer1 输入时钟预分频比选择位
11 = 1:256
10 = 1:64
01 = 1:8
00 = 1:1

bit 3　**未实现**：读为 0

bit 2　**TSYNC**：Timer1 外部时钟输入同步选择位
当 TCS = 1 时：
1 = 同步外部时钟输入
0 = 不同步外部时钟输入
当 TCS = 0 时：
该位为无关位。

bit 1　**TCS**：Timer1 时钟源选择位
1 = 来自 T1CK 引脚的外部时钟 （上升沿触发计数）
0 = 内部时钟 （FCY）

bit 0　**未实现**：读为 0

© 2007 Microchip Technology Inc.

12.0 TIMER2/3、TIMER4/5、TIMER6/7 和 TIMER8/9

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10 器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

Timer2/3、Timer4/5、Timer6/7 和 Timer8/9 模块为 32 位定时器，也可被配置为 8 个具有可选工作模式的独立 16 位定时器。

作为 32 位定时器，Timer2/3、Timer4/5、Timer6/7 和 Timer8/9 具有三种工作模式：

- 具有所有 16 位工作模式（异步计数器模式除外）的两个独立 16 位定时器（例如，Timer2 和 Timer3）
- 单个 32 位定时器
- 单个 32 位同步计数器

还支持以下功能：

- 定时器门控操作
- 可选择的预分频比设置
- 空闲和休眠模式下的定时器工作
- 在 32 位周期寄存器匹配时产生中断
- 输入捕捉和输出比较模块的时基（仅限 Timer2 和 Timer3）
- ADC1 事件触发器（仅限 Timer2/3）
- ADC2 事件触发器（仅限 Timer4/5）

所有 8 个 16 位定时器都能作为同步定时器或计数器。它们也提供上面所列的功能，但事件触发功能除外，它仅由 Timer2/3 实现。通过设置 T2CON、T3CON、T4CON、T5CON、T6CON、T7CON、T8CON 和 T9CON 寄存器中的相应位来确定工作模式和使能特性。T2CON、T4CON、T6CON 和 T8CON 在寄存器 12-1 中作了一般介绍。T3CON、T5CON、T7CON 和 T9CON 如寄存器 12-2 所示。

对于32位定时器/计数器工作，Timer2、Timer4、Timer6或Timer8是32位定时器的低位字，而Timer3、Timer5、Timer7 或 Timer9 是高位字。

注： 对于 32 位工作，T3CON、T5CON、T7CON 和 T9CON 控制位将被忽略。设置和控制只使用 T2CON、T4CON、T6CON 和 T8CON 控制位。32 位定时器模块采用 Timer2、Timer4、Timer6 和 Timer8 时钟和门控输入，但中断由 Timer3、Timer5、Timer7 和 Timer9 中断标志位产生。

要为 32 位工作配置 Timer2/3、Timer4/5、Timer6/7 或 Timer8/9：

1. 将相应的 T32 控制位置 1。
2. 使用 TCKPS<1:0> 位为 Timer2、Timer4、Timer6 或 Timer8 选择预分频比。
3. 使用相应的 TCS 和 TGATE 位设置时钟和门控模式。
4. 装入定时器的周期值。PR3、PR5、PR7 或 PR9 包含值的高位字，而 PR2、PR4、PR6 或 PR8 包含低位字。
5. 如果需要中断，将中断允许位 T3IE、T5IE、T7IE 或 T9IE 置 1。使用优先级位 T3IP<2:0>、T5IP<2:0>、T7IP<2:0> 或 T9IP<2:0> 来设置中断优先级。Timer2、Timer4、Timer6 或 Timer8 控制定时器，而中断由 Timer3、Timer5、Timer7 或 Timer9 产生。
6. 将相应的 TON 位置 1。

任意时刻定时器的值被存储在寄存器对 TMR3:TMR2、TMR5:TMR4、TMR7:TMR6 或 TMR9:TMR8 中。TMR3、TMR5、TMR7 或 TMR9 总是包含计数值的高位字，而 TMR2、TMR4、TMR6 或 TMR8 包含低位字。

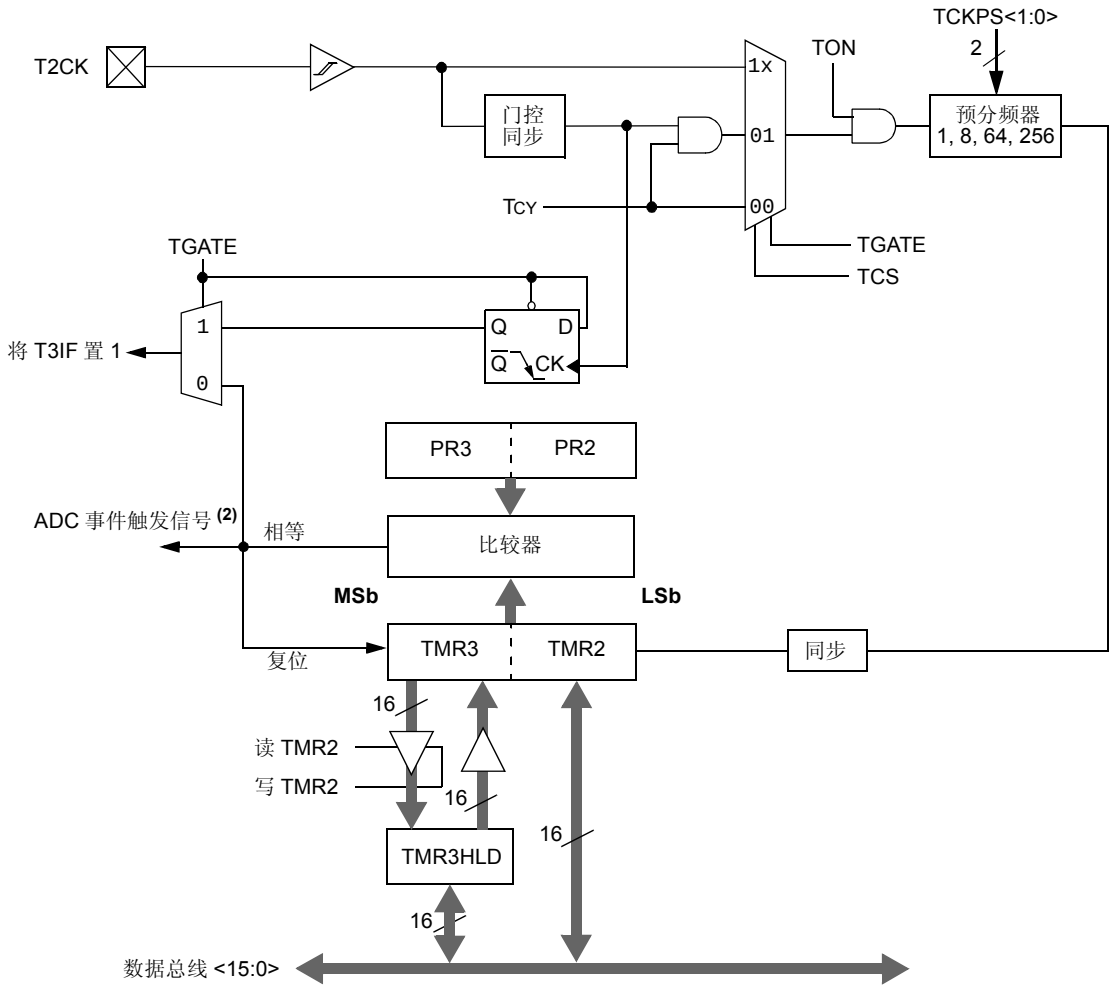
要将任一定时器配置为独立的 16 位工作：

1. 清零与该定时器对应的 T32 位。
2. 使用 TCKPS<1:0> 位选择定时器预分频比。
3. 使用 TCS 和 TGATE 位设置时钟和门控模式。
4. 将定时器的周期值装入 PRx 寄存器。
5. 如果需要中断，将中断允许位 TxIE 置 1。使用优先级位 TxIP<2:0> 来设置中断优先级。
6. 将 TON 位置 1。

图 12-1 给出了 32 位定时器对（Timer4/5）的框图示例，图 12-2 给出了工作在 16 位模式下的定时器（Timer4）示例。

注： 只有 Timer2 和 Timer3 能触发 DMA 数据传输。

图 12-1： TIMER2/3 （32 位）框图 (1)

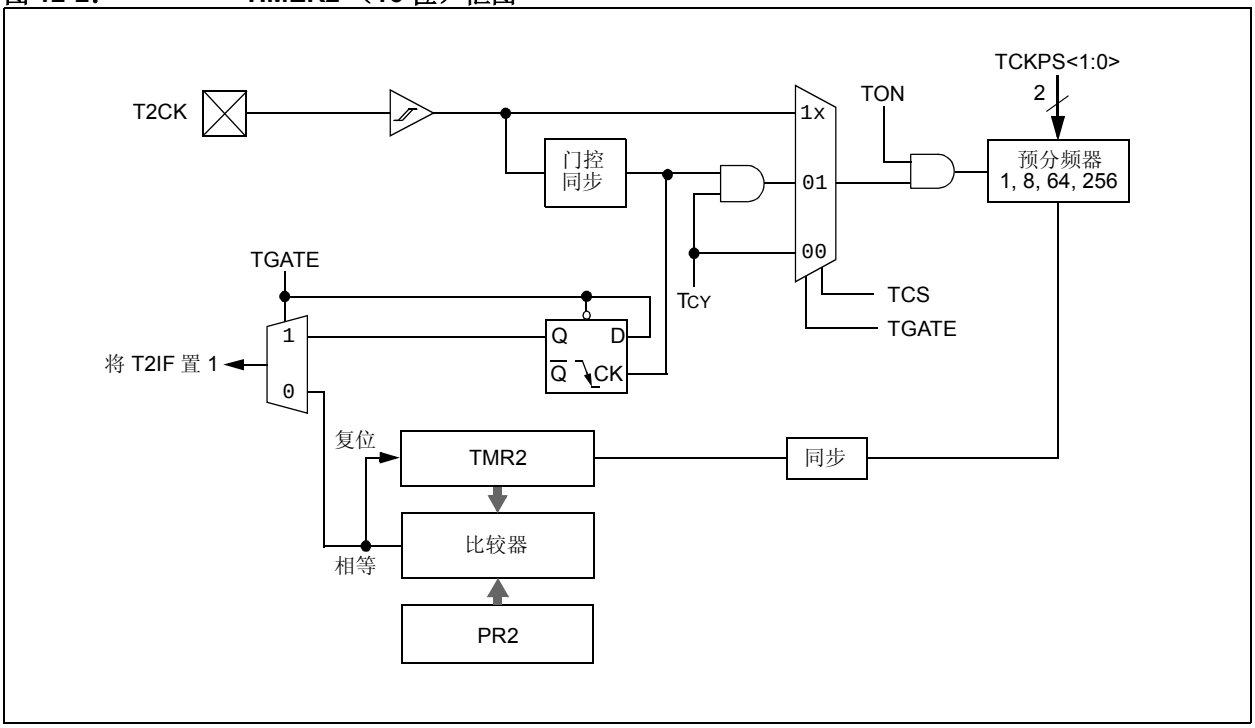


注 1： 必须将 32 位定时器控制位 T32 置 1，以使定时器工作在 32 位定时器 / 计数器模式。所有控制位都与 T2CON 寄存器相对应。

2： ADC 事件触发功能仅可在 Timer2/3 上使用。

© 2007 Microchip Technology Inc.

图 12-2：　　TIMER2 （16 位）框图



寄存器 12-1： TxCON （T2CON、 T4CON、 T6CON 或 T8CON）控制寄存器

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON	—	TSIDL	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0
—	TGATE	TCKPS<1:0>		T32(1)	—	TCS	—
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **TON**：Timerx 使能位
当 T32 = 1 时：
1 = 启动 32 位 Timerx/y
0 = 停止 32 位 Timerx/y
当 T32 = 0 时：
1 = 启动 16 位 Timerx
0 = 停止 16 位 Timerx

bit 14 **未实现**：读为 0

bit 13 **TSIDL**：空闲模式停止位
1 = 当器件进入空闲模式时，模块停止工作
0 = 在空闲模式下模块继续工作

bit 12-7 **未实现**：读为 0

bit 6 **TGATE**：Timerx 门控时间累加使能位
当 TCS = 1 时：
该位为无关位。
当 TCS = 0 时：
1 = 使能门控时间累加
0 = 禁止门控时间累加

bit 5-4 **TCKPS<1:0>**：Timerx 输入时钟预分频比选择位
11 = 1:256
10 = 1:64
01 = 1:8
00 = 1:1

bit 3 **T32**：32 位定时器模式选择位 (1)
1 = Timerx 和 Timery 形成一个 32 位定时器
0 = Timerx 和 Timery 作为两个 16 位定时器

bit 2 **未实现**：读为 0

bit 1 **TCS**：Timerx 时钟源选择位
1 = 来自 TxCK 引脚的外部时钟 （上升沿触发计数）
0 = 内部时钟 （FCY）

bit 0 **未实现**：读为 0

注 1： 在 32 位模式下， T3CON 控制位不影响 32 位定时器的工作。

© 2007 Microchip Technology Inc.

寄存器 12-2： TyCON （T3CON、 T5CON、 T7CON 或 T9CON）控制寄存器

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON(1)	—	TSIDL(1)	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	U-0
—	TGATE(1)	TCKPS<1:0>(1)		—	—	TCS(1)	—
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **TON：**Timery 使能位 (1)
1 = 启动 16 位 Timery
0 = 停止 16 位 Timery

bit 14 **未实现：**读为 0

bit 13 **TSIDL：**空闲模式停止位 (1)
1 = 当器件进入空闲模式时，模块停止工作
0 = 在空闲模式下模块继续工作

bit 12-7 **未实现：**读为 0

bit 6 **TGATE：**Timery 门控时间累加使能位 (1)
当 TCS = 1 时：
该位为无关位。
当 TCS = 0 时：
1 = 使能门控时间累加
0 = 禁止门控时间累加

bit 5-4 **TCKPS<1:0>：**Timer3 输入时钟预分频比选择位 (1)
11 = 1:256
10 = 1:64
01 = 1:8
00 = 1:1

bit 3-2 **未实现：**读为 0

bit 1 **TCS：**Timery 时钟源选择位 (1)
1 = 来自 TyCK 引脚的外部时钟 （上升沿触发计数）
0 = 内部时钟 （FCY）

bit 0 **未实现：**读为 0

注 1： 当使能 32 位工作 （T2CON<3> = 1）时，这些位对 Timery 的工作没有影响；所有定时器功能都通过 T2CON 进行设置。

注：

© 2007 Microchip Technology Inc.

13.0 输入捕捉

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10 器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

输入捕捉模块在需要频率（周期）和脉冲测量的应用中很有用。 dsPIC33FJXXXGPX06/X08/X10 器件支持最多 8 路输入捕捉通道。

当 ICx 引脚上有事件发生时，输入捕捉模块捕捉选定时基寄存器的 16 位值。导致发生捕捉的事件分为以下三类：

1. 简单捕捉事件模式
   -每当 ICx 引脚上的输入信号出现下降沿时捕捉定时器值
   -每当 ICx 引脚上的输入信号出现上升沿时捕捉定时器值
2. 在每个边沿（上升沿和下降沿）都捕捉定时器值
3. 预分频捕捉事件模式
   -ICx 引脚上的输入信号每出现 4 个上升沿捕捉一次定时器值
   -ICx 引脚上的输入信号每出现 16 个上升沿捕捉一次定时器值

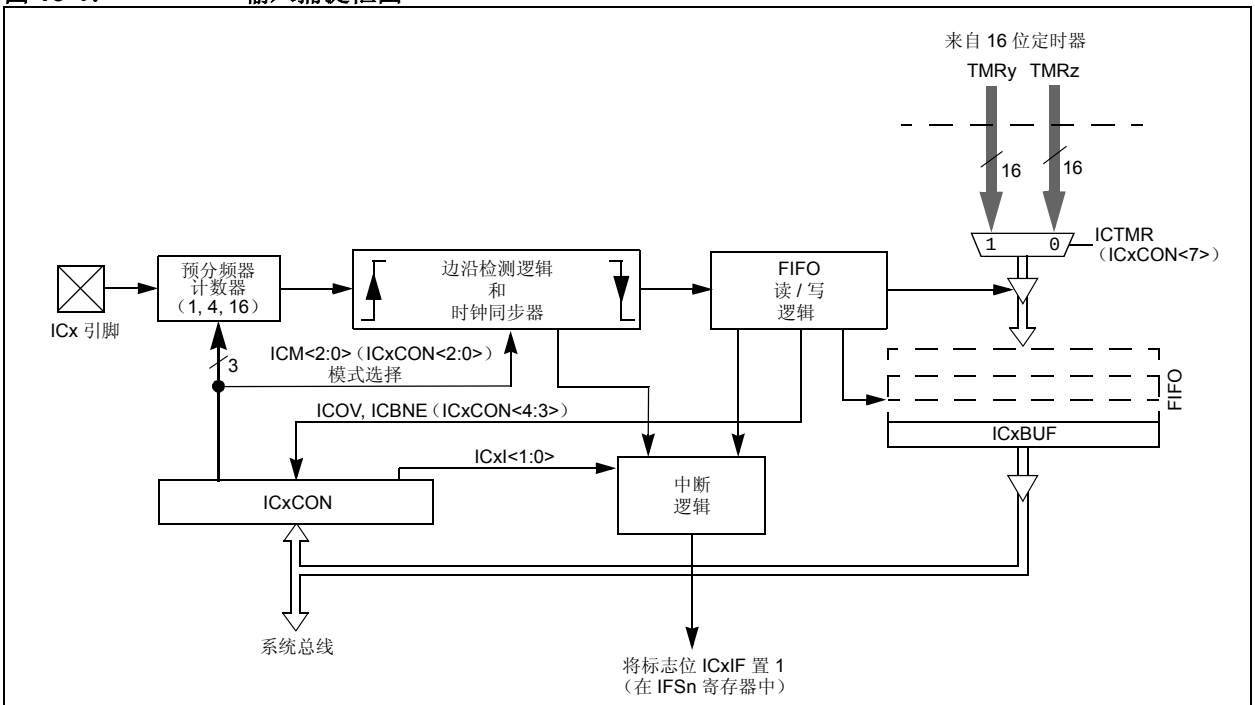
每路输入捕捉通道都可以选择两个16位定时器（Timer2 或 Timer3）之一作为时基。选定定时器可以使用内部或外部时钟。

其他工作特性包括：

- 当 CPU 在休眠和空闲模式时通过捕捉引脚上的信号将器件唤醒
- 输入捕捉事件中断
- 用于存储捕捉值的 4 字 FIFO 缓冲器
  - 可选择在填充完 1、 2、 3 或 4 个缓冲单元后产生中断
- 输入捕捉也可用来提供额外的外部中断源

注： 只有 IC1 和 IC2 能触发 DMA 数据传输。如果需要进行 DMA 数据传输， FIFO 缓冲区大小必须设置为 1 （ICI<1:0> = 00）。

图 13-1： 输入捕捉框图



注： 在信号、寄存器或位名称中的“x”表示捕捉通道的编号。

13.1 输入捕捉寄存器

寄存器 13-1： ICxCON：输入捕捉 x 控制寄存器

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	—	ICSIDL	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R-0, HC	R-0, HC	R/W-0	R/W-0	R/W-0
ICTMR(1)	ICI<1:0>		ICOV	ICBNE	ICM<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-14 **未实现：**读为 0

bit 13 **ICSIDL：**输入捕捉模块在空闲模式下停止的控制位
1 = 在 CPU 空闲模式下输入捕捉模块将停止工作
0 = 在 CPU 空闲模式下输入捕捉模块将继续工作

bit 12-8 **未实现：**读为 0

bit 7 **ICTMR：**输入捕捉定时器选择位 (1)
1 = 发生捕捉事件时捕捉 TMR2 的内容
0 = 发生捕捉事件时捕捉 TMR3 的内容

bit 6-5 **ICI<1:0>：**选择每次发生中断捕捉的次数的位
11 = 每 4 次捕捉事件中断一次
10 = 每 3 次捕捉事件中断一次
01 = 每 2 次捕捉事件中断一次
00 = 每次捕捉事件中断一次

bit 4 **ICOV：**输入捕捉溢出状态标志位 （只读）
1 = 发生了输入捕捉溢出
0 = 未发生输入捕捉溢出

bit 3 **ICBNE：**输入捕捉缓冲器空状态位 （只读）
1 = 输入捕捉缓冲器非空，至少可以再读一次捕捉值
0 = 输入捕捉缓冲器为空

bit 2-0 **ICM<2:0>：**输入捕捉模式选择位
111 = 当器件处于休眠或空闲模式时，输入捕捉通道仅用作中断引脚 （只检测上升沿，所有其他控制位都不适用。）
110 = 未使用 （模块被禁止）
101 = 捕捉模式，每 16 个上升沿捕捉一次
100 = 捕捉模式，每 4 个上升沿捕捉一次
011 = 捕捉模式，每个上升沿捕捉一次
010 = 捕捉模式，每个下降沿捕捉一次
001 = 捕捉模式，每个边沿（上升沿和下降沿）捕捉一次（ICI<1:0> 位不控制该模式下的中断产生。）
000 = 输入捕捉模块关闭

注 1： 定时器选择可能会有变化。请参见器件数据手册了解详细信息。

© 2007 Microchip Technology Inc.

14.0 输出比较

注：	本数据手册总结了dsPIC33FJXXXGPX06/X08/X10 器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

14.1 设置产生单个输出脉冲

当 OCM 控制位（OCxCON<2:0>）被设置为 100 时，所选的输出比较通道将OCx引脚初始化为低电平状态并产生单输出脉冲。

若要产生单输出脉冲，需要遵循以下步骤（这些步骤假设定时器源在开始时是关闭的，但这并不是对模块工作的要求）：

1. 确定指令时钟周期时间。考虑定时器源的外部时钟频率（如果使用）和定时器预分频比的设置。
2. 计算从 TMRy 起始值（0000h）到输出脉冲的上升沿所需的时间。
3. 根据所需的脉冲宽度和到脉冲上升沿的时间计算出现脉冲下降沿的时间。
4. 将以上步骤 2 和步骤 3 中计算出的值分别写入输出比较寄存器 OCxR 和输出比较辅助寄存器 OCxRS。
5. 将定时器周期寄存器 PRy 的值设置为等于或大于输出比较辅助寄存器 OCxRS 中的值。
6. 将 OCM 位设置为 100，并将 OCTSEL（OCxCON<3>）位设置为要求的定时器源。此时 OCx 引脚状态被驱动为低电平。
7. 将 TON（TyCON<15>）位设置为 1 以使比较时基计数。
8. 在 TMRy 和 OCxR 第一次匹配时，OCx 引脚将被驱动为高电平。
9. 当递增定时器TMRy和输出比较辅助寄存器OCxRS发生匹配时，在 OCx 引脚上驱动脉冲的第二个边沿（即下降沿）。OCx 引脚上不会驱动输出额外的脉冲，OCx 引脚将保持为低电平。第二次比较匹配事件会导致 OCxIF 中断标志位置 1。如果已通过将 OCxIE 位置 1 允许中断，将产生中断。关于外设中断的更多信息，请参见**第 6.0 节 “中断控制器”**。
10. 要发出另一个单脉冲输出，在需要的情况下要更改定时器和比较寄存器的设置，然后执行写操作，将 OCM 位设置为 100。不需要禁止和重新使能定时器以及将 TMRy 寄存器清零，但这样做可能对确定一个已知事件时间边界发出的脉冲有好处。

在输出脉冲下降沿后不一定要禁止输出比较模块。重写 OCxCON 寄存器的值可以发出另一个脉冲。

14.2 设置产生连续输出脉冲

当 OCM 控制位（OCxCON<2:0>）被设置为 101 时，所选的输出比较通道在每次比较匹配事件发生时将OCx引脚初始化为低电平状态并输出脉冲。

用户若要将模块配置为产生连续的输出脉冲流，需要遵循以下步骤（这些步骤假设定时器源在开始时是关闭的，但这并不是对模块工作的要求）：

1. 确定指令时钟周期时间。考虑定时器源的外部时钟频率（如果使用）和定时器预分频比的设置。
2. 计算从 TMRy 起始值（0000h）到输出脉冲的上升沿所需的时间。
3. 根据所需的脉冲宽度和到脉冲上升沿的时间计算出现脉冲下降沿的时间。
4. 将以上步骤 2 和步骤 3 中计算出的值分别写入输出比较寄存器 OCxR 和输出比较辅助寄存器 OCxRS。
5. 将定时器周期寄存器 PRy 的值设置为等于或大于输出比较辅助寄存器 OCxRS 中的值。
6. 将 OCM 位设置为 101，并将 OCTSEL 位设置为要求的定时器源。此时 OCx 引脚状态被驱动为低电平。
7. 通过将 TON（TyCON<15>）位设置为 1 使能比较时基。
8. 在 TMRy 和 OCxR 第一次匹配时，OCx 引脚将被驱动为高电平。
9. 当比较时基 TMRy 和输出比较辅助寄存器 OCxRS 发生匹配时，OCx 引脚上出现脉冲的第二个边沿（即下降沿）。
10. 第二次比较匹配事件会导致 OCxIF 中断标志位置 1。
11. 当比较时基和相应的定时器周期寄存器中的值匹配时，TMRy 寄存器复位为 0x0000 并重新开始计数。
12. 重复步骤 8 到步骤 11，可无限制地产生连续脉冲流。OCxIF 标志位在每次 OCxRS 与 TMRy 的比较匹配事件发生时置 1。

14.3 脉宽调制模式

当将输出比较模块配置为 PWM 操作时，需要遵循以下步骤：

1. 通过写所选的定时器周期寄存器 （PRy）设置 PWM 周期。
2. 通过写 OCxRS 寄存器设置 PWM 占空比。
3. 向 OxCR 寄存器中写入初始占空比。
4. 如果需要的话，允许定时器和输出比较模块的中断。如果要使用 PWM 故障引脚，则必须设置输出比较中断。
5. 通过写输出比较模式位 OCM<2:0> （OCxCON<2:0>），将输出比较模块配置为两种 PWM 工作模式中的一种。
6. 通过设置 TON （TxCON<15>） = 1 设置 TMRy 预分频值并使能时基。

注： 在第一次使能输出比较模块之前，必须先初始化 OCxR 寄存器。当模块工作于 PWM 模式时， OCxR 寄存器变为只读的占空比寄存器。 OCxR 中保存的值成为第一个 PWM 周期的 PWM 占空比。直到时基周期匹配发生，输出比较辅助寄存器 OCxRS 的内容才会被传输到 OCxR。

14.3.1 PWM 周期

PWM 周期可通过写入 PRy（定时器周期寄存器）来指定。 PWM 周期可由公式 14-1 计算：

公式 14-1： 计算 PWM 周期

PWM 周期 = [(PRy) + 1] • TCY • (定时器预分频值)
其中：
PWM 频率 = 1/[PWM 周期]

注： 如果 PRy 的值为 N，则会使 PWM 周期为 N + 1 个时基计数周期。例如，如果写入 PRy 寄存器的值为 7，则将产生由 8 个时基周期组成的 PWM 周期。

14.3.2 PWM 占空比

PWM 占空比是通过写 OCxRS 寄存器指定的。可以在任何时候写 OCxRS 寄存器，但是在 PRy 和 TMRy 发生匹配 （即周期结束）前占空比值不会被锁存到 OCxR 中。这可以为 PWM 占空比提供双重缓冲，对于 PWM 的无毛刺操作是极其重要的。在 PWM 模式下，OCxR 是只读寄存器。

PWM 占空比有一些重要的边界参数，包括：

- 如果输出比较寄存器 OCxR 中装入 0000h，则 OCx 引脚将保持低电平 （占空比为 0%）。
- 如果 OCxR 大于 PRy （定时器周期寄存器），则引脚将保持高电平 （占空比为 100%）。
- 如果 OCxR 等于 PRy，则 OCx 引脚在一个时基计数周期内为低电平，而在其余所有的计数周期内均为高电平。

请参见例 14-1 了解 PWM 模式时序的详细信息。表 14-1 给出了器件工作速度为 10 MIPS 时的 PWM 频率和分辨率示例。

公式 14-2： 计算最大 PWM 分辨率

$$\text{最大 PWM 分辨率（位）} = \frac{\log_{10}\left(\frac{FCY}{FPWM}\right)}{\log_{10}(2)} \text{ 位}$$

例 14-1： PWM 周期和占空比计算

1. 在期望 PWM 频率为 52.08 kHz， FCY = 16 MHz 且 Timer2 预分频比为 1:1 时，计算定时器周期寄存器的值。
   TCY = 62.5 ns
   PWM 周期 = 1/PWM 频率 = 1/52.08 kHz = 19.2 µs
   PWM 周期 = (PR2 + 1) • TCY • (Timer2 预分频值)
   19.2 µs = (PR2 + 1) • 62.5 ns • 1
   PR2 = 306
2. 在 PWM 频率为 52.08 kHz 且器件的时钟速率为 32 MHz 时，计算占空比的最大分辨率。
   PWM 分辨率 = $\log_{10}(FCY/FPWM)/\log_{10}2)$ 位
   = $(\log_{10}(16\text{ MHz}/52.08\text{ kHz})/\log_{10}2)$ 位
   = 8.3 位

© 2007 Microchip Technology Inc.

表 14-1：　　器件工作在 4 MIPS （FCY = 4 MHz）时的 PWM 频率和分辨率示例

PWM 频率	7.6 Hz	61 Hz	122 Hz	977 Hz	3.9 kHz	31.3 kHz	125 kHz
定时器预分频比	8	1	1	1	1	1	1
周期寄存器的值	FFFFh	FFFFh	7FFFh	0FFFh	03FFh	007Fh	001Fh
分辨率 （位）	16	16	15	12	10	7	5

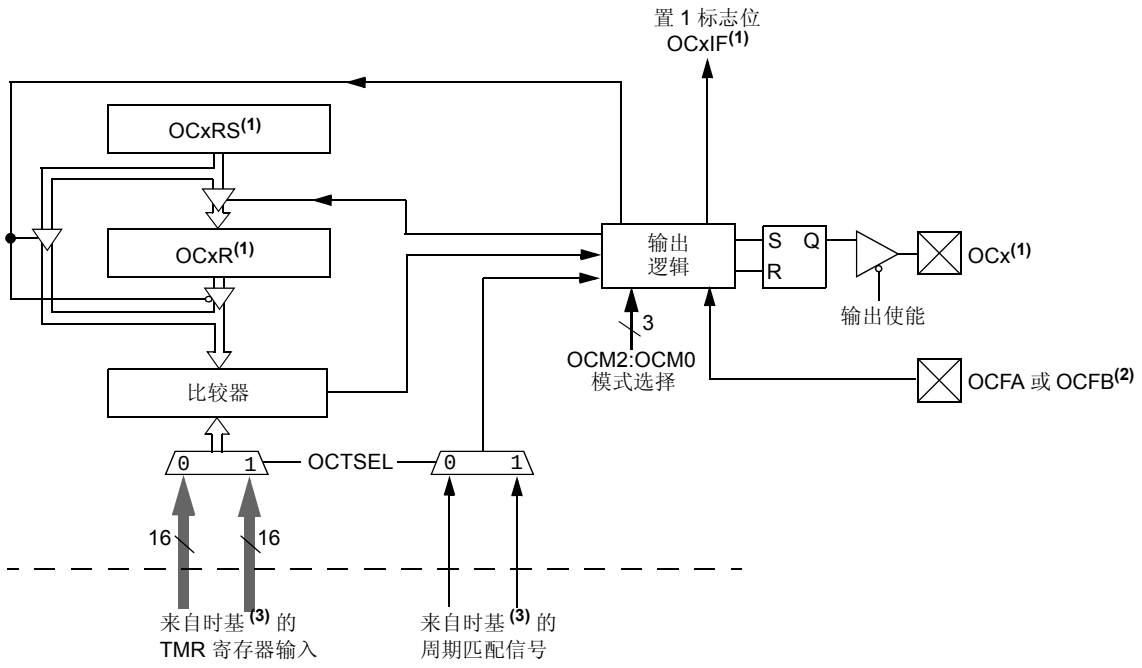
表 14-2：　　器件工作在 16 MIPS （FCY = 16 MHz）时的 PWM 频率和分辨率示例

PWM 频率	30.5 Hz	244 Hz	488 Hz	3.9 kHz	15.6 kHz	125 kHz	500 kHz
定时器预分频比	8	1	1	1	1	1	1
周期寄存器的值	FFFFh	FFFFh	7FFFh	0FFFh	03FFh	007Fh	001Fh
分辨率 （位）	16	16	15	12	10	7	5

表 14-3：　　器件工作在 40 MIPS （FCY = 40 MHz）时的 PWM 频率和分辨率示例

PWM 频率	76 Hz	610 Hz	1.22 Hz	9.77 kHz	39 kHz	313 kHz	1.25 MHz
定时器预分频比	8	1	1	1	1	1	1
周期寄存器的值	FFFFh	FFFFh	7FFFh	0FFFh	03FFh	007Fh	001Fh
分辨率 （位）	16	16	15	12	10	7	5

图 14-1：　　输出比较模块框图



注　1： 显示 “x” 之处与各自输出比较通道 1 到 8 相关的寄存器对应。
2： OCFA 引脚控制 OC1 到 OC4 通道。 OCFB 引脚控制 OC5 到 OC8 通道。
3： 每个输出比较通道可以使用两个可选时基中的一个。关于与该模块相关的时基，请参见器件数据手册。

注：　只有 OC1 和 OC2 能触发 DMA 数据传输。

对应的 TRISx 位必须清零以将相关 I/O 引脚配置为输出比较输出。

14.4 输出比较寄存器

寄存器 14-1： OCxCON：输出比较 x 控制寄存器

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	—	OCSIDL	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R-0 HC	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	OCFLT	OCTSEL(1)	OCM<2:0>		
bit 7							bit 0

图注：	HC = 用硬件清零	HS = 用硬件置 1	
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-14 **未实现：**读为 0

bit 13 **OCSIDL：**在空闲模式下停止输出比较控制位
1 = 输出比较 x 将在 CPU 空闲模式下停止
0 = 输出比较 x 将在 CPU 空闲模式下继续工作

bit 12-5 **未实现：**读为 0

bit 4 **OCFLT：**PWM 故障条件状态位
1 = 产生了 PWM 故障条件 （仅可用硬件清零）
0 = 未产生 PWM 故障条件
（仅当 OCM<2:0> = 111 时，才使用该位。）

bit 3 **OCTSEL：**输出比较定时器选择位 (1)
1 = Timer3 是比较 x 的时钟源
0 = Timer2 是比较 x 的时钟源

bit 2-0 **OCM<2:0>：**输出比较模式选择位
111 = OCx 处于 PWM 模式，使能故障引脚
110 = OCx 处于 PWM 模式，禁止故障引脚
101 = 初始化 OCx 引脚为低电平，在 OCx 引脚上产生连续输出脉冲
100 = 初始化 OCx 引脚为低电平，在 OCx 引脚上产生单个输出脉冲
011 = 比较事件使 OCx 引脚的电平翻转
010 = 初始化 OCx 引脚为高电平，比较事件强制 OCx 引脚为低电平
001 = 初始化 OCx 引脚为低电平，比较事件强制 OCx 引脚为高电平
000 = 禁止输出比较通道

注 1： 关于输出比较模块可用的特定时基，请参见器件数据手册。

© 2007 Microchip Technology Inc.

15.0 串行外设接口 （SPI）

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

串行外设接口 （Serial Peripheral Interface， SPI）模块是用于同其他外设或单片机器件进行通信的同步串行接口。这些外设可以是串行 EEPROM、移位寄存器、显示驱动器和 ADC 等。SPI 模块与 Motorola® 的 SPI 和 SIOP 兼容。

注： 在本章中，SPI 模块统称为 SPIx，或分别称为 SPI1 和 SPI2。特殊功能寄存器也使用类似的符号表示。例如，SPIxCON 指 SPI1 或 SPI2 模块的控制寄存器。

每个 SPI 模块由一个用于将数据移入和移出的 16 位移位寄存器 SPIxSR （其中 x = 1 或 2）和一个缓冲寄存器 SPIxBUF 组成。控制寄存器 SPIxCON 用来配置模块。另外，还有一个状态寄存器 SPIxSTAT 用来表明各种状态条件。

串行接口由 4 个引脚组成：SDIx （串行数据输入）、SDOx（串行数据输出）、SCKx（移位时钟输入或输出）和 $\overline{SSx}$ （低电平有效从选择）。

在主模式下工作时，SCK 是时钟输出，但在从模式下它是时钟输入。

一组 8 或 16 个时钟脉冲将数据位从 SPIxSR 移出到 SDOx 引脚，同时将 SDIx 引脚的数据位移入 SPIxSR。当传输完成时产生一个中断，相应的中断标志位 （SPI1IF 或 SPI2IF）置 1。通过中断允许位（SPI1IE 或 SPI2IE）可以禁止该中断。

接收操作是双重缓冲的。当接收完字节时，将字节从 SPIxSR 送到 SPIxBUF。

当从 SPIxSR 传输新数据到 SPIxBUF 时，如果接收缓冲器已满，模块会将 SPIROV 位置 1，表明产生了溢出条件。数据从 SPIxSR 向 SPIxBUF 的传输不会完成，新数据将丢失。当 SPIROV 为 1 时，模块将不会对 SCKx 电平的跳变进行响应；实际上会禁止模块，直到用户软件读 SPIxBUF 为止。

发送写操作同样也是双重缓冲的。用户写入 SPIxBUF。当主或从传输完成后，移位寄存器 （SPIxSR）的内容将被移入接收缓冲器。如果已经向缓冲寄存器写了任何发送数据，发送缓冲器的内容将被移入 SPIxSR。于是，接收到的数据存放在 SPIxBUF 中，而 SPIxSR 中的发送数据已经准备就绪，可进行下一次传输。

注： 发送缓冲器 （SPIxTXB）和接收缓冲器（SPIxRXB）都映射到相同的寄存器地址 SPIxBUF。请勿对 SPIxBUF 寄存器执行读 - 修改 - 写操作 （例如针对位的指令）。

要将 SPI 模块设置为工作在主模式下：

1. 如果使用中断：
   a) 将相应 IFSn 寄存器中的 SPIxIF 位清零。
   b) 将相应 IECn 寄存器中的 SPIxIE 位置 1。
   c) 通过写相应 IPCn 寄存器中的 SPIxIP 位来设置中断优先级。
2. 将要求的设置写入 SPIxCON 寄存器，且 MSTEN （SPIxCON1<5>） = 1。
3. 将 SPIROV 位 （SPIxSTAT<6>）清零。
4. 通过将 SPIEN 位 （SPIxSTAT<15>）置 1 使能 SPI 工作。
5. 将待发送数据写入 SPIxBUF 寄存器。发送 （和接收）在数据写入 SPIxBUF 寄存器时立即开始。

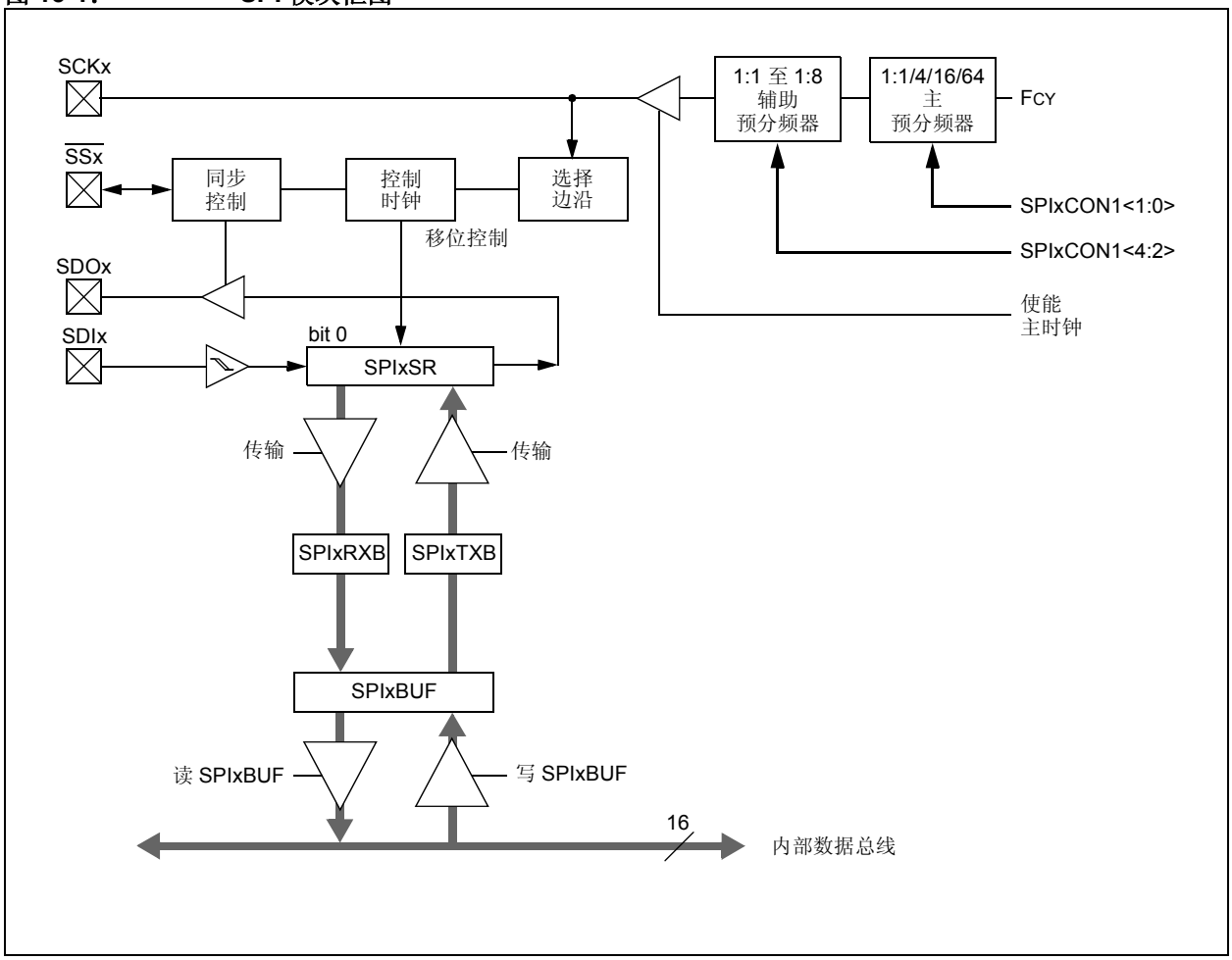
要将 SPI 模块设置为工作在从模式下：

1. 将 SPIxBUF 寄存器清零。
2. 如果使用中断：
   a) 将相应 IFSn 寄存器中的 SPIxIF 位清零。
   b) 将相应 IECn 寄存器中的 SPIxIE 位置 1。
   c) 通过写相应 IPCn 寄存器中的 SPIxIP 位来设置中断优先级。
3. 将要求的设置写入 SPIxCON1 和 SPIxCON2 寄存器，且 MSTEN （SPIxCON1<5>） = 0。
4. 将 SMP 位清零。
5. 如果 CKE 位置 1，则必须将 SSEN 位（SPIxCON1<7>）置 1 来使能 $\overline{SSx}$ 引脚。
6. 将 SPIROV 位 （SPIxSTAT<6>）清零。
7. 通过将 SPIEN 位 （SPIxSTAT<15>）置 1 使能 SPI 工作。

SPI 模块会产生一个中断表明字节或字传输完成，还会为所有的 SPI 错误条件产生单独的中断。

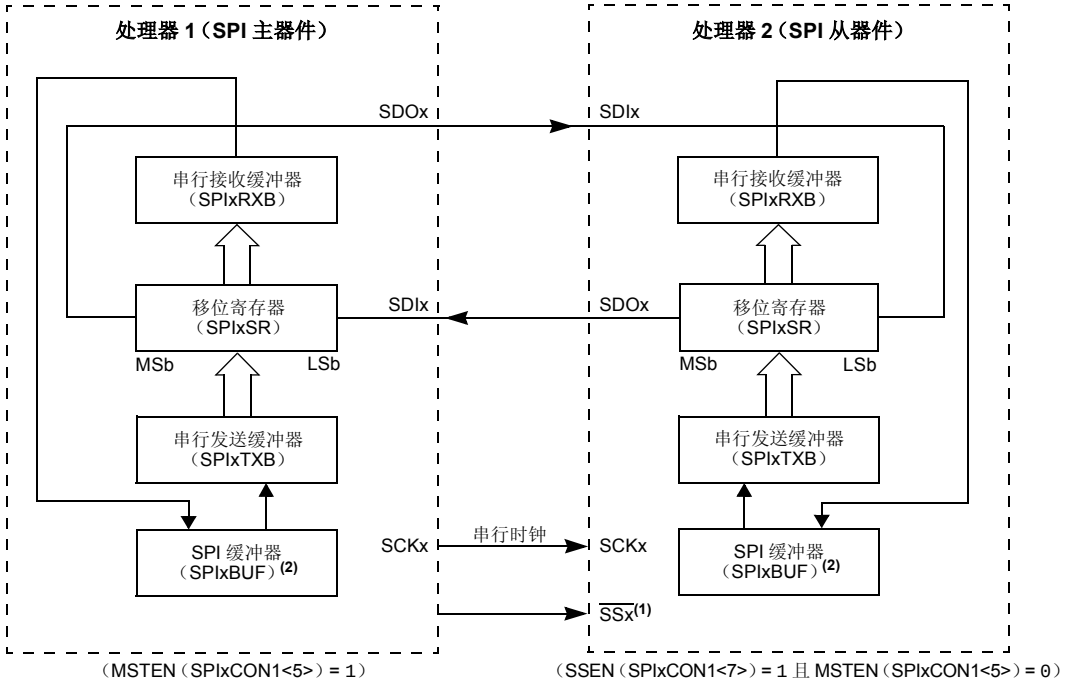
注：	SPI1 和 SPI2 都能触发 DMA 数据传输。如果 SPI1 或 SPI2 被选作 DMA IRQ 源，在由于 SPI1 或 SPI2 字节或字传输导致 SPI1IF 或 SPI2IF 位置 1 时，会发生 DMA 传输。

图 15-1： SPI 模块框图



© 2007 Microchip Technology Inc.

图 15-2：　　SPI 主 / 从连接



注　1： 可选择在从模式下使用 $\overline{SSx}$ 引脚。
2： 用户必须向 SPIxBUF 写入发送数据，或从 SPIxBUF 读取接收数据。 SPIxTXB 和 SPIxRXB 寄存器被存储器映射到 SPIxBUF。

图 15-3：　　SPI 主 - 帧主模式连接图

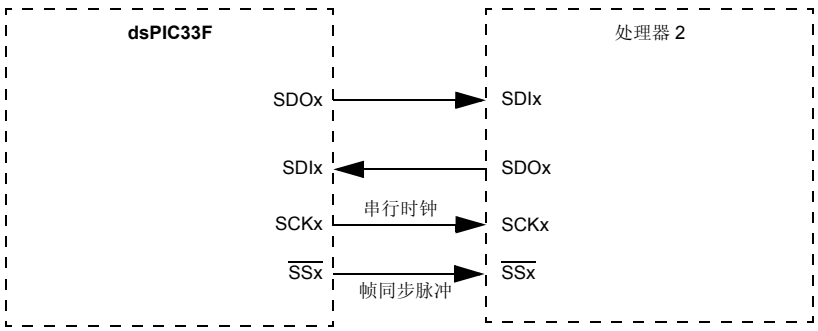


图 15-4：　　SPI 主 - 帧从模式连接图

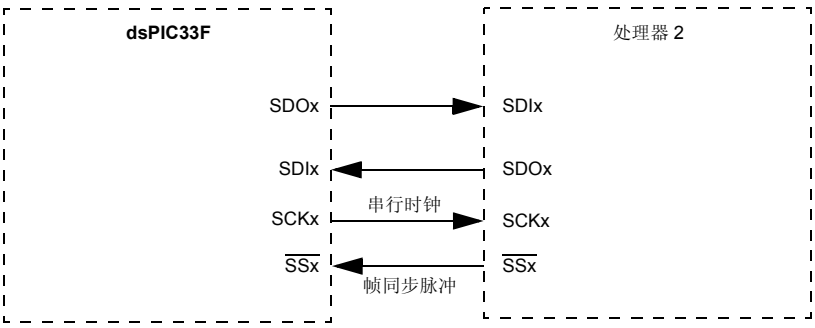


图 15-5：　　SPI 从 - 帧主模式连接图

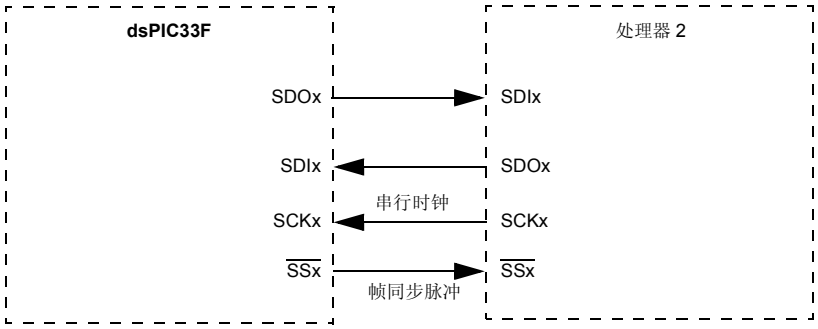
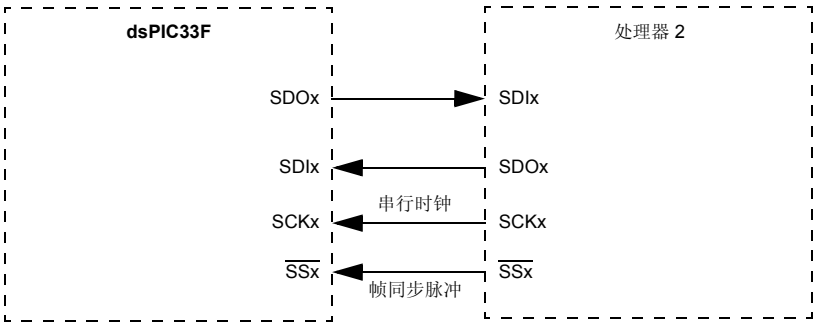


图 15-6：　　SPI 从 - 帧从模式连接图



公式 15-1：　　器件工作频率和 SPI 时钟速度之间的关系

$$F_{SCK} = \frac{F_{CY}}{\text{主预分频比} * \text{辅助预分频比}}$$

表 15-1：　　SCKx 频率示例

FCY = 40 MHz		**辅助预分频比设置**				
		1:1	**2:1**	**4:1**	**6:1**	**8:1**
主预分频比设置	1:1	无效	无效	10000	6666.67	5000
	4:1	10000	5000	2500	1666.67	1250
	16:1	2500	1250	625	416.67	312.50
	64:1	625	312.5	156.25	104.17	78.125
FCY = 5 MHz						
主预分频比设置	1:1	5000	2500	1250	833	625
	4:1	1250	625	313	208	156
	16:1	313	156	78	52	39
	64:1	78	39	20	13	10

注：　表中 SCKx 频率的单位为 kHz。

© 2007 Microchip Technology Inc.

寄存器 15-1： SPIxSTAT：SPIx 状态和控制寄存器

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
SPIEN	—	SPISIDL	—	—	—	—	—
bit 15							bit 8

U-0	R/C-0	U-0	U-0	U-0	U-0	R-0	R-0
—	SPIROV	—	—	—	—	SPITBF	SPIRBF
bit 7							bit 0

图注：	C = 可清零位		
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **SPIEN**：SPIx 使能位
1 = 使能模块并将 SCKx、 SDOx、 SDIx 和 $\overline{SSx}$ 配置为串口引脚
0 = 禁止模块

bit 14 **未实现**：读为 0

bit 13 **SPISIDL**：空闲模式停止位
1 = 当器件进入空闲模式时，模块停止工作
0 = 在空闲模式下模块继续工作

bit 12-7 **未实现**：读为 0

bit 6 **SPIROV**：接收溢出标志位
1 = 一个新字节 / 字已完全接收并丢弃。用户软件没有读先前保存在 SPIxBUF 寄存器中的数据。
0 = 未发生溢出

bit 5-2 **未实现**：读为 0

bit 1 **SPITBF**：SPIx 发送缓冲器满状态位
1 = 发送尚未开始， SPIxTXB 为满
0 = 发送已开始， SPIxTXB 为空
当 CPU 写 SPIxBUF 存储单元并装入 SPIxTXB 时，该位由硬件自动置 1。
当 SPIx 模块将数据从 SPIxTXB 传输到 SPIxSR 时，该位由硬件自动清零。

bit 0 **SPIRBF**：SPIx 接收缓冲器满状态位
1 = 接收完成， SPIxRXB 为满
0 = 接收未完成， SPIxRXB 为空
当 SPIx 将数据从 SPIxSR 传输到 SPIxRXB 时，该位由硬件自动置 1。
当内核通过读 SPIxBUF 存储单元读 SPIxRXB 时，该位由硬件自动清零。

寄存器 15-2： SPIxCON1：SPIx 控制寄存器 1

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE(1)
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SSEN	CKP	MSTEN	SPRE<2:0>			PPRE<1:0>	
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-13 **未实现：**读为 0

bit 12 **DISSCK：**禁止 SCKx 引脚位 （仅限 SPI 主模式）
1 = 禁止内部 SPI 时钟，引脚用作 I/O
0 = 使能内部 SPI 时钟

bit 11 **DISSDO：**禁止 SDOx 引脚位
1 = 模块不使用 SDOx 引脚；引脚用作 I/O
0 = SDOx 引脚由模块控制

bit 10 **MODE16：**字 / 字节通信选择位
1 = 采用字宽 （16 位）通信
0 = 采用字节宽 （8 位）通信

bit 9 **SMP：**SPIx 数据输入采样相位位
主模式：
1 = 在数据输出时间的末端采样输入数据
0 = 在数据输出时间的中间采样输入数据
从模式：
当在从模式下使用 SPIx 时，必须将 SMP 清零。

bit 8 **CKE：**SPIx 时钟边沿选择位 (1)
1 = 串行输出数据在时钟从工作状态转变为空闲状态时变化 （见 bit 6）
0 = 串行输出数据在时钟从空闲状态转变为工作状态时变化 （见 bit 6）

bit 7 **SSEN：**从选择使能位 （从模式）
1 = SSx 引脚用于从模式
0 = SSx 引脚不被模块使用。引脚由端口功能控制。

bit 6 **CKP：**时钟极性选择位
1 = 空闲状态时时钟信号为高电平；工作状态时为低电平
0 = 空闲状态时时钟信号为低电平；工作状态时为高电平

bit 5 **MSTEN：**主模式使能位
1 = 主模式
0 = 从模式

bit 4-2 **SPRE<2:0>：**辅助预分频比位 （主模式）
111 = 辅助预分频比 1:1
110 = 辅助预分频比 2:1
•
•
•
000 = 辅助预分频比 8:1

bit 1-0 **PPRE<1:0>：**主预分频比位 （主模式）
11 = 主预分频比 1:1
10 = 主预分频比 4:1
01 = 主预分频比 16:1
00 = 主预分频比 64:1

注 1： 在帧 SPI 模式下不使用 CKE 位。在帧 SPI 模式 （FRMEN = 1）下，用户应将该位编程为 0。

© 2007 Microchip Technology Inc.

寄存器 15-3： SPIxCON2：SPIx 控制寄存器 2

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
FRMEN	SPIFSD	FRMPOL	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	U-0
—	—	—	—	—	—	FRMDLY	—
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **FRMEN**：帧 SPIx 支持位
1 = 使能帧 SPIx 支持 （$\overline{SSx}$ 引脚用作帧同步脉冲输入 / 输出）
0 = 禁止帧 SPIx 支持

bit 14 **SPIFSD**：帧同步脉冲方向控制位
1 = 帧同步脉冲输入 （从器件）
0 = 帧同步脉冲输出 （主器件）

bit 13 **FRMPOL**：帧同步脉冲极性位
1 = 帧同步脉冲为高电平有效
0 = 帧同步脉冲为低电平有效

bit 12-2 **未实现**：读为 0

bit 1 **FRMDLY**：帧同步脉冲边沿选择位
1 = 帧同步脉冲与第一个位时钟一致
0 = 帧同步脉冲比第一个位时钟提前

bit 0 **未实现**：禁止使用用户应用程序将该位设置为 1。

注：

© 2007 Microchip Technology Inc.

16.0 I²C

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10 器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

I²C 模块（16 位接口）为 I²C 串行通信标准的从模式和多主器件模式提供完全的硬件支持。

dsPIC33FJXXXGPX06/X08/X10 器件具有最多两个 I²C 接口模块，分别用 I2C1 和 I2C2 表示。每个 I²C 模块具有一个双引脚接口：两个引脚分别是时钟引脚 SCLx 和数据引脚 SDAx。

每个 I²C 模块 “x”（x = 1 或 2）提供以下主要特性：

- I²C 接口支持主 / 从工作模式。
- I²C 从模式支持 7 位和 10 位地址。
- I²C 主模式支持 7 位和 10 位地址。
- I²C 端口允许主器件和从器件之间的双向传输。
- I²C 端口的串行时钟同步可以用作握手机制来暂停和继续串行传输 （SCLREL 控制）。
- I²C 支持多主器件工作；检测总线冲突并相应地进行仲裁。

16.1 工作模式

硬件完全实现了 I²C 标准和快速模式规范的所有主从功能，以及 7 位和 10 位寻址。

在 I²C 总线上， I²C 模块既可作为主器件工作，也可作为从器件工作。

支持 I²C 模块工作在以下模式：

- 带 7 位地址的 I²C 从模式
- 带 10 位地址的 I²C 从模式
- 带 7 位或 10 位地址的 I²C 主模式

关于每种模式的通信序列的详细信息，请参见《dsPIC33F 系列参考手册》。

16.2 I²C 寄存器

I2CxCON 和 I2CxSTAT 分别是控制寄存器和状态寄存器。I2CxCON 寄存器是可读写的。I2CxSTAT 的低 6 位是只读的，它的其余位是可读写的。

I2CxRSR 是用于移动数据的移位寄存器，而 I2CxRCV 是从 / 向其中读写数据字节的缓冲寄存器。I2CxRCV 是接收缓冲器。I2CxTRN 是发送寄存器，发送操作期间，字节将会写入 I2CxTRN。

I2CxADD 寄存器存放从地址。状态位 ADD10 表明是否为 10 位地址模式。 I2CxBRG 用来保存波特率发生器（Baud Rate Generation， BRG）的重载值。

在进行接收时，I2CxRSR 和 I2CxRCV 一起形成双重缓冲接收器。当 I2CxRSR 接收到一个完整的字节后，字节被送至 I2CxRCV 并产生中断脉冲。

16.3 I²C 中断

I²C 模块产生两个中断标志：MI2CxIF（I²C 主事件中断标志）和 SI2CxIF（I²C 从事件中断标志）。在所有 I²C 错误条件下都会产生一个单独的中断。

16.4 波特率发生器

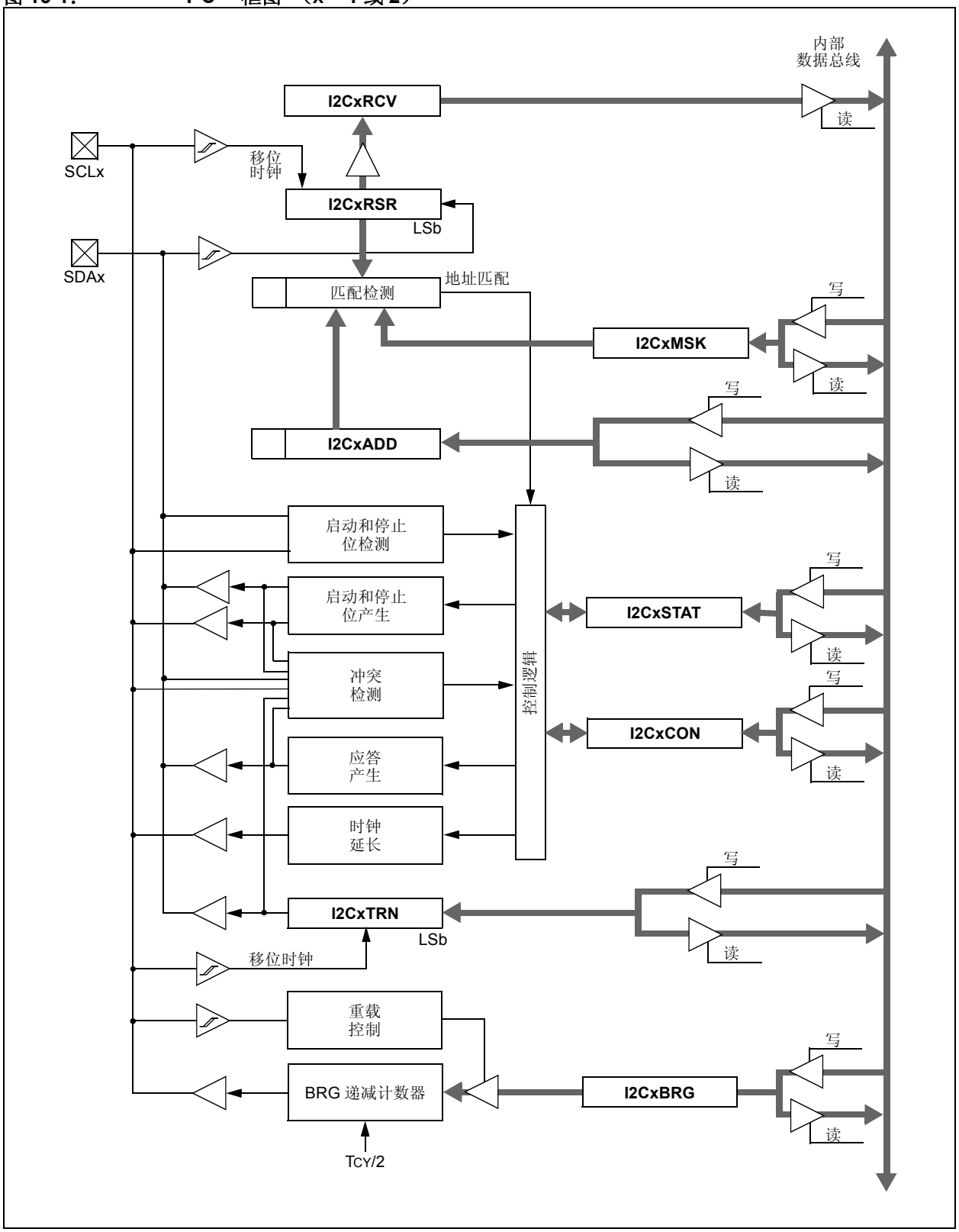
在 I²C 主模式下，BRG 的重载值位于 I2CxBRG 寄存器中。 BRG 装入该值后， BRG 将递减计数到 0 并停止，直到发生再次装入。如果发生时钟仲裁，例如 SCLx 引脚采样为高电平时，将重载 BRG。

根据 I²C 标准，FSCL 可以是 100 kHz 或 400 kHz。然而，用户可以指定任何波特率，最高到 1 MHz。I2CxBRG 的值不能是 0 或 1。

公式 16-1： 串行时钟速率

$$I2CxBRG = \left(\frac{FCY}{FSCL} - \frac{FCY}{10{,}000{,}00}\right) - 1$$

图 16-1：　　I²C™ 框图 （x = 1 或 2）



© 2007 Microchip Technology Inc.

16.5 I²C 模块地址

I2CxADD 寄存器包含从模式地址。这是一个 10 位寄存器。

如果 A10M 位 （I2CxCON<10>）为 0，则模块把地址解释为 7 位地址。当接收到地址时，将收到的地址与 I2CxADD 寄存器中的低 7 位进行比较。

如果 A10M 位为 1，则将地址视为 10 位地址。当接收到地址时，将收到的地址与二进制值“11110 A9 A8”（其中 A9 和 A8 是 I2CxADD 的高 2 位）进行比较。如果值匹配，下一个地址将与 I2CxADD 的低 8 位进行比较，这是 10 位寻址协议规定的。

表 16-1： dsPIC33FJXXXGPX06/X08/X10 支持的 7 位 I²C™ 从地址

0x00	广播呼叫地址或启动字节
0x01-0x03	保留
0x04-0x07	Hs 模式主机码
0x08-0x77	有效 7 位地址
0x78-0x7b	有效 10 位地址 （低 7 位）
0x7c-0x7f	保留

16.6 从地址掩码

I2CxMSK 寄存器 （寄存器 16-3）将 7 位和 10 位地址模式下地址中的某些位指定为“无关位”。将 I2CxMSK 寄存器中某个特定位置 1（= 1），不论相应的地址位的值是 0 还是 1，工作在从模式下的模块都会作出响应。例如，当将 I2CxMSK 设置为 00100000 时，工作在从模式下的模块将检测两个地址0000000和00100000。

要使能地址掩码，必须通过将IPMIEN位（I2CxCON<11>）清零来禁止智能外设管理接口 （Intelligent Peripheral Management Interface， IPMI）。

16.7 IPMI 支持

控制位 IPMIEN 允许模块支持智能外设管理接口（IPMI）。当此位置 1 时，模块将接受并对所有地址执行操作。

16.8 广播呼叫地址支持

广播呼叫地址能寻址所有器件。当使用此地址时，理论上所有器件都应该以应答信号作出响应。

广播呼叫地址是 I²C 协议为特定目的保留的 8 个地址之一。它由全 0 地址位组成，且 R_W = 0。

当广播呼叫使能 （GCEN）位置 1（I2CxCON<7> = 1）时，将识别广播呼叫地址。在响应中断时，通过读 I2CxRCV 的内容可以检测到中断的来源，从而确定地址是特定于器件的还是广播呼叫地址。

16.9 自动时钟延长

在从模式下，模块可以通过时钟延长同步读缓冲器和写主器件。

16.9.1 发送时钟延长

在 10 位和 7 位发送模式下，如果 TBF 位清零（表明缓冲器为空），则在第 9 个时钟的下降沿后将 SCLREL 位清零，就实现了时钟延长。

在从发送模式下，始终进行时钟延长，与 STREN 位无关。在允许继续发送之前，用户 ISR 必须将 SCLREL 位置 1。通过使 SCLx 线保持为低电平，在主器件启动另一个发送序列之前，用户就有时间执行 ISR 并装入 I2CxTRN 的内容。

16.9.2 接收时钟延长

在从接收模式下，I2CxCON 寄存器中的 STREN 位可用来使能时钟延长。当 STREN 位置 1 时， SCLx 引脚将在每个数据接收序列结束时保持为低电平。

在允许继续接收之前，用户 ISR 必须将 SCLREL 位置 1。通过使 SCLx 线保持为低电平，在主器件启动另一个接收序列之前，用户就有时间执行ISR并读出I2CxRCV的内容。这可以防止缓冲器溢出。

16.10 软件控制的时钟延长 （STREN = 1）

当 STREN 位为 1 时，可以用软件清零 SCLREL 位，从而实现对时钟延长的软件控制。

如果 STREN 位为 0，软件对 SCLREL 位的写操作将被忽略，不会对 SCLREL 位产生影响。

16.11 斜率控制

对于快速模式 （400 kHz）， I^2C 标准要求对 SDAx 和 SCLx 信号进行斜率控制。如果需要的话，用户可通过控制位 DISSLW 来禁止斜率控制。对于 1 MHz 模式，必须禁止斜率控制。

16.12 时钟仲裁

在任何接收、发送或重复启动 / 停止条件期间，当主器件释放了 SCLx 引脚 （允许 SCLx 悬空为高电平）时，就会发生时钟仲裁。当允许SCLx引脚悬空为高电平时，波特率发生器（BRG）暂停计数，直到 SCLx 引脚被实际采样到高电平为止。当SCLx引脚被采样到高电平时，波特率发生器重新装入 I2CxBRG 的内容并开始计数。当外部器件使时钟保持为低电平时，这可以始终保证SCLx高电平时间至少为一个BRG计满返回计数周期。

16.13 多主器件通信、总线冲突和总线仲裁

通过总线仲裁实现了对多主器件模式的支持。当主器件在 SDAx 引脚上输出地址 / 数据位时，如果一个主器件使 SDAx 悬空为高电平从而输出一个 1，而另一个主器件要输出 0，就会发生仲裁。当 SCLx 引脚悬空为高电平时，数据应该是稳定的。如果 SDAx 上预期数据是 1，但从 SDAx 引脚采样到的数据是 0，那么就发生了总线冲突。主器件将 I^2C 主事件中断标志位置 1，并且将 I^2C 端口的主器件部分复位到空闲状态。

© 2007 Microchip Technology Inc.

寄存器 16-1： I2CxCON：I2Cx 控制寄存器

R/W-0	U-0	R/W-0	R/W-1 HC	R/W-0	R/W-0	R/W-0	R/W-0
I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0 HC	R/W-0 HC	R/W-0 HC	R/W-0 HC	R/W-0 HC
GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

图注：	U = 未实现位，读为 0		
R = 可读位	W = 可写位	HS = 用硬件置 1	HC = 用硬件清零
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **I2CEN**：I2Cx 使能位
1 = 使能 I2Cx 模块，并将 SDAx 和 SCLx 引脚配置为串口引脚
0 = 禁止 I2Cx 模块。所有 I^2C 引脚由端口功能控制。

bit 14 **未实现**：读为 0

bit 13 **I2CSIDL**：空闲模式停止位
1 = 当器件进入空闲模式时，模块停止工作
0 = 在空闲模式下模块继续工作

bit 12 **SCLREL**：SCLx 释放控制位 （作为 I^2C 从器件工作时）
1 = 释放 SCLx 时钟
0 = 保持 SCLx 时钟为低电平 （时钟延长）

如果 STREN = 1：
该位可读可写 （即软件可以写入 0 来启动时钟延长或写入 1 来释放时钟）。在从器件发送开始时由硬件清零。在从器件接收结束时由硬件清零。

如果 STREN = 0：
该位可读且可被置 1 （即软件只能写入 1 来释放时钟）。在从器件发送开始时由硬件清零。

bit 11 **IPMIEN**：智能外设管理接口 （IPMI）使能位
1 = 使能 IPMI 模式；应答所有地址
0 = 禁止 IPMI 模式

bit 10 **A10M**：10 位从器件地址位
1 = I2CxADD 为 10 位从器件地址
0 = I2CxADD 为 7 位从器件地址

bit 9 **DISSLW**：禁止斜率控制位
1 = 禁止斜率控制
0 = 使能斜率控制

bit 8 **SMEN**：SMBus 输入电平位
1 = 使能符合 SMBus 规范的 I/O 引脚门限值
0 = 禁止 SMBus 输入门限值

bit 7 **GCEN**：广播呼叫使能位 （作为 I^2C 从器件工作时）
1 = 允许在 I2CxRSR 接收到广播呼叫地址时产生中断 （已使能模块接收）
0 = 禁止广播呼叫地址

bit 6 **STREN**：SCLx 时钟延长使能位 （作为 I^2C 从器件工作时）
与 SCLREL 位配合使用。
1 = 使能软件或接收时钟延长
0 = 禁止软件或接收时钟延长

寄存器 16-1：　　I2CxCON：I2Cx 控制寄存器 （续）

bit 5　**ACKDT**：应答数据位 （作为 I^2C 主器件工作时，适用于主器件接收过程）
当软件启动应答序列时将发送的值。
1 = 在应答时发送 NACK
0 = 在应答时发送 ACK

bit 4　**ACKEN**：应答序列使能位 （作为 I^2C 主器件工作时，适用于主器件接收过程）
1 = 在 SDAx 和 SCLx 引脚上发出应答序列，并发送 ACKDT 数据位。
在主器件应答序列结束时由硬件清零。
0 = 应答序列不在进行中

bit 3　**RCEN**：接收使能位 （作为 I^2C 主器件工作时）
1 = 使能 I^2C 接收模式。在主器件接收完数据字节的第 8 位时由硬件清零。
0 = 接收序列不在进行中

bit 2　**PEN**：停止条件使能位 （作为 I^2C 主器件工作时）
1 = 在 SDAx 和 SCLx 引脚上发出停止条件。在主器件停止序列结束时由硬件清零。
0 = 停止条件不在进行中

bit 1　**RSEN**：重复启动条件使能位 （作为 I^2C 主器件工作时）
1 = 在 SDAx 和 SCLx 引脚上发出重复启动条件。在主器件重复启动序列结束时由硬件清零。
0 = 重复启动条件不在进行中

bit 0　**SEN**：启动条件使能位 （作为 I^2C 主器件工作时）
1 = 在 SDAx 和 SCLx 引脚上发出启动条件。在主器件启动序列结束时由硬件清零。
0 = 启动条件不在进行中

© 2007 Microchip Technology Inc.

寄存器 16-2： I2CxSTAT：I2Cx 状态寄存器

R-0 HSC	R-0 HSC	U-0	U-0	U-0	R/C-0 HS	R-0 HSC	R-0 HSC
ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10
bit 15							bit 8

R/C-0 HS	R/C-0 HS	R-0 HSC	R/C-0 HSC	R/C-0 HSC	R-0 HSC	R-0 HSC	R-0 HSC
IWCOL	I2COV	D_A	P	S	R_W	RBF	TBF
bit 7							bit 0

图注：	U = 未实现位，读为 0		
R = 可读位	W = 可写位	HS = 用硬件置 1	HSC = 用硬件置 1/ 清零
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **ACKSTAT**：应答状态位 （作为 I^2C 主器件工作时，适用于主器件发送操作）
1 = 接收到来自从器件的 NACK
0 = 接收到来自从器件的 ACK
在从器件应答结束时由硬件置 1 或清零。

bit 14 **TRSTAT**：发送状态位 （作为 I^2C 主器件工作时，适用于主器件发送操作）
1 = 主器件正在进行发送 （8 位 + ACK）
0 = 主器件不在进行发送
在主器件发送开始时由硬件置 1。在从器件应答结束时由硬件清零。

bit 13-11 **未实现**：读为 0

bit 10 **BCL**：主器件总线冲突检测位
1 = 主器件工作期间检测到了总线冲突
0 = 未发生冲突
检测到总线冲突时由硬件置 1。

bit 9 **GCSTAT**：广播呼叫状态位
1 = 接收到广播呼叫地址
0 = 未接收到广播呼叫地址
当地址与广播呼叫地址匹配时由硬件置 1。当检测到停止条件时由硬件清零。

bit 8 **ADD10**：10 位地址状态位
1 = 10 位地址匹配
0 = 10 位地址不匹配
当与匹配的 10 位地址的第 2 个字节匹配时由硬件置 1。当检测到停止条件时由硬件清零。

bit 7 **IWCOL**：写冲突检测位
1 = 因为 I^2C 模块忙，尝试写 I2CxTRN 寄存器失败
0 = 未发生冲突
当总线忙时写 I2CxTRN 会使硬件将该位置 1 （由软件清零）。

bit 6 **I2COV**：接收溢出标志位
1 = 当 I2CxRCV 寄存器仍然保存原先的字节时接收到了新字节
0 = 未溢出
尝试将数据从 I2CxRSR 传输到 I2CxRCV 时由硬件置 1 （由软件清零）。

bit 5 **D_A**：数据 / 地址位 （作为 I^2C 从器件工作时）
1 = 表示上次接收的字节为数据
0 = 表示上次接收的字节为器件地址
器件地址匹配时由硬件清零。在作为从器件接收到数据字节时由硬件置 1。

bit 4 **P**：停止位
1 = 表示上一次检测到停止位
0 = 上一次未检测到停止位
当检测到启动、重复启动或停止条件时由硬件置 1 或清零。

寄存器 16-2：　　I2CxSTAT：I2Cx 状态寄存器 （续）

bit 3　　**S**：启动位
1 = 表示上次检测到启动 （或重复启动）位
0 = 上一次未检测到起始位
当检测到启动、重复启动或停止条件时由硬件置 1 或清零。

bit 2　　**R_W**：读 / 写信息位 （作为 I^2C 从器件工作时）
1 = 读——表示数据传输自从器件输出
0 = 写——表示数据传输输入到从器件
接收到 I^2C 器件地址字节后由硬件置 1 或清零。

bit 1　　**RBF**：接收缓冲器满状态位
1 = 接收完成， I2CxRCV 为满
0 = 接收未完成， I2CxRCV 为空
用接收到的字节写 I2CxRCV 时由硬件置 1。用软件读 I2CxRCV 时由硬件清零。

bit 0　　**TBF**：发送缓冲器满状态位
1 = 发送正在进行中， I2CxTRN 为满
0 = 发送完成， I2CxTRN 为空
用软件写 I2CxTRN 时由硬件置 1。数据发送完成时由硬件清零。

© 2007 Microchip Technology Inc.

寄存器 16-3：　　I2CxMSK：I2Cx 从模式地址掩码寄存器

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	AMSK9	AMSK8
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
AMSK7	AMSK6	AMSK5	AMSK4	AMSK3	AMSK2	AMSK1	AMSK0
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-10　**未实现：**读为 0

bit 9-0　**AMSKx：**地址位 x 的掩码选择位
1 = 使能输入报文的地址中位 x 的掩码；在此位置上不需要位匹配
0 = 禁止位 x 的掩码；在此位置上需要位匹配

注：

© 2007 Microchip Technology Inc.

17.0 通用异步收发器 （UART）

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10 器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

通用异步收发器（Universal Asynchronous Receiver Transmitter， UART）模块是 dsPIC33FJXXXGPX06/X08/X10 器件系列提供的串行 I/O 模块之一。UART 是可以与外设（例如个人电脑、LIN、RS-232 和 RS-485 接口）通信的全双工异步系统。模块还通过 UxCTS 和 UxRTS 引脚支持硬件流控制选项，其中还包括 IrDA® 编码器和解码器。

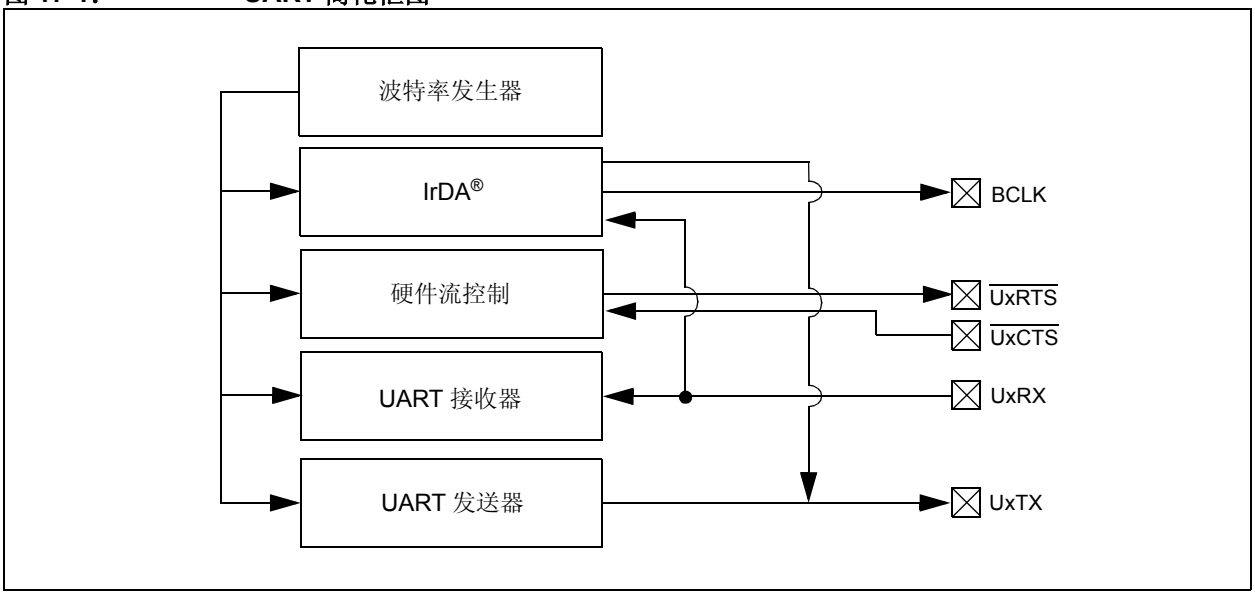
UART 模块的主要特性有：

- 通过UxTX和UxRX引脚进行全双工8位或9位数据传输
- 偶校验、奇校验或无奇偶校验选项 （对于 8 位数据）
- 一个或两个停止位
- 通过 UxCTS 和 UxRTS 引脚支持硬件流控制选项
- 完全集成的波特率发生器，具有 16 位预分频器
- 当器件工作在 16 MIPS 时，波特率范围从 1 Mbps 到 15 Mbps
- 4 级深度先进先出 （First-In-First-Out， FIFO）发送数据缓冲器
- 4 级深度 FIFO 接收数据缓冲器
- 奇偶校验、帧和缓冲器溢出错误检测
- 支持带地址检测的 9 位模式 （第 9 位 = 1）
- 发送和接收中断
- 所有 UART 错误条件下可分别产生中断
- 用于诊断支持的环回模式
- 支持同步和间隔字符
- 支持自动波特率检测
- IrDA 编码器和解码器逻辑
- 用于 IrDA 支持的 16 倍频波特率时钟输出

图 17-1 给出了 UART 的简化框图。 UART 模块由以下至关重要的硬件组件组成：

- 波特率发生器
- 异步发送器
- 异步接收器

图 17-1： UART 简化框图



注 1： UART1 和 UART2 都能触发 DMA 数据传输。如果 U1TX、U1RX、U2TX 或 U2RX 被选作 DMA IRQ 源，在由于 UART1 或 UART2 的发送或接收导致 U1TXIF、 U1RXIF、 U2TXIF 或 U2RXIF 位置 1 时，会发生 DMA 传输。

2： 如果需要 DMA 传输，必须将 UART 发送 / 接收 FIFO 缓冲器的大小设置为 1 字节 / 字（即 UTXISEL<1:0> = 00 且 URXISEL<1:0> = 00）。

17.1 UART 波特率发生器 （BRG）

UART 模块包含一个专用的 16 位波特率发生器。BRGx 寄存器控制一个自由运行的 16 位定时器的周期。公式 17-1 给出了 BRGH = 0 时计算波特率的公式。

公式 17-1： UART 波特率 （BRGH = 0）

$$\text{波特率} = \frac{FCY}{16 \bullet (BRGx + 1)}$$

$$BRGx = \frac{FCY}{16 \bullet \text{波特率}} - 1$$

注： FCY 表示指令周期时钟频率 （FOSC/2）。

例 17-1 给出了如下条件下的波特率误差计算：

- FCY = 4 MHz
- 目标波特率 = 9600

最大可能波特率（BRGH = 0）是 FCY/16（当 BRGx = 0 时），最小可能波特率是 FCY/(16 * 65536)。

公式 17-2 给出 BRGH = 1 时计算波特率的公式。

公式 17-2： UART 波特率 （BRGH = 1）

$$\text{波特率} = \frac{FCY}{4 \bullet (BRGx + 1)}$$

$$BRGx = \frac{FCY}{4 \bullet \text{波特率}} - 1$$

注： FCY 表示指令周期时钟频率 （FOSC/2）。

最大可能波特率（BRGH = 1）是 FCY/4（当 BRGx = 0 时），最小可能波特率是 FCY/(4 * 65536)。

向 BRGx 寄存器写入新值会使 BRG 定时器复位 （清零）。这可以确保 BRG 无需等待定时器溢出就可以产生新的波特率。

例 17-1： 波特率误差计算 （BRGH = 0）

目标波特率	=	FCY/(16 (BRGx + 1))
求解 BRGx 值：		
BRGx	=	((FCY/ 目标波特率)/16) – 1
BRGx	=	((4000000/9600)/16) – 1
BRGx	=	25
计算波特率	=	4000000/(16 (25 + 1))
	=	9615
误差	=	(计算波特率 – 目标波特率)/目标波特率
	=	(9615 – 9600)/9600
	=	0.16%

© 2007 Microchip Technology Inc.

17.2 8 位数据模式下的发送

1. 设置 UART：
   a) 将适当的值写入数据位、奇偶校验位和停止位。
   b) 将适当的波特率值写入 BRGx 寄存器。
   c) 设置发送和接收中断允许位和优先级位。
2. 使能 UART。
3. 将 UTXEN 位置 1 （产生发送中断）。
4. 将数据字节写入 UxTXREG 字的低字节。该数据字节将被立即传输给发送移位寄存器 （TSR），且在波特率时钟的下一个上升沿开始移出串行比特流。
5. 或者，当 UTXEN = 0 时，数据字节也可被发送，且随后用户可将 UTXEN 置 1。由于波特率时钟将从清零状态启动，这将立即开始发送串行比特流。
6. 中断控制位 UTXISEL<1:0> 的设置决定何时产生发送中断。

17.3 9 位数据模式下的发送

1. 设置 UART （如**第 17.2 节 “8 位数据模式下的发送”**所述）。
2. 使能 UART。
3. 将 UTXEN 位置 1 （产生发送中断）。
4. 仅向 UxTXREG 写入一个 16 位的值。
5. 向 UxTXREG 写入一个字可触发 9 位数据向 TSR 的传输。串行比特流将会在波特率时钟的第一个上升沿开始移出。
6. 中断控制位 UTXISEL<1:0> 的设置决定何时产生发送中断。

17.4 间隔和同步发送序列

下述序列会发送一个报文帧头，包括一个间隔字符和其后的一个自动波特率同步字节。

1. 将 UART 配置为所需的模式。
2. 将 UTXEN 和 UTXBRK 置 1——设置间隔字符。
3. 将一个“虚拟”字符装入 UxTXREG 寄存器中以启动发送 （值被忽略）。
4. 向 UxTXREG 写入 0x55——将同步字符装入发送 FIFO 中。
5. 间隔字符发送后，硬件会将 UTXBRK 位复位。然后开始发送同步字符。

17.5 8 位或 9 位数据模式下的接收

1. 设置 UART （如**第 17.2 节 “8 位数据模式下的发送”**所述）。
2. 使能 UART。
3. 当接收到一个或多个数据字符时，将会根据中断控制位 URXISEL<1:0> 的设置产生接收中断。
4. 读 OERR 位以确定是否发生了溢出错误。OERR 位必须用软件复位。
5. 读 UxRXREG。

读取 UxRXREG 字符的行为会将下一个字符传送到接收 FIFO 的顶部，其中包含一组新的 PERR 和 FERR 值。

17.6 使用 $\overline{\text{UxCTS}}$ 和 $\overline{\text{UxRTS}}$ 引脚的流控制

UARTx 允许发送 （$\overline{\text{UxCTS}}$）和 UARTx 请求发送（$\overline{\text{UxRTS}}$）是两个与 UART 模块相关、由硬件控制的低电平有效引脚。这两个引脚允许 UART 运行在单工模式和流控制模式下。它们用于控制数据终端设备 （Data Terminal Equipment，DTE）之间的发送和接收。UxMODE 寄存器中的 UEN<1:0> 位用来配置这两个引脚。

17.7 红外支持

UART 模块提供两种类型的红外 UART 支持：

- IrDA 时钟输出支持外部 IrDA 编码器和解码器 （传统模块支持）
- 完全实现了 IrDA 编码器和解码器

17.7.1 外部 IrDA 支持——IrDA 时钟输出

为了支持外部 IrDA 编码器和解码器，可将 BCLK 引脚（和 UxRTS 引脚相同）配置为产生 16 倍频的波特率时钟。当使能了 UART 模块且 UEN<1:0> = 11 时，BCLK 引脚将输出 16 倍频的波特率时钟，用于支持 IrDA 编解码器芯片。

17.7.2 内置 IrDA 编码器和解码器

UART 模块在其内部完全实现了 IrDA 编码器和解码器。内置 IrDA 编码器和解码器的功能可通过 IREN 位（UxMODE<12>）来使能。当使能（IREN = 1）时，接收引脚 （UxRX）可作为红外接收器的输入引脚。发送引脚 （UxTX）可作为红外发送器的输出引脚。

寄存器 17-1：　UxMODE：UARTx 模式寄存器

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0(2)	R/W-0(2)
UARTEN	—	USIDL	IREN(1)	RTSMD	—	UEN<1:0>	
bit 15							bit 8

R/W-0 HC	R/W-0	R/W-0 HC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WAKE	LPBACK	ABAUD	URXINV	BRGH	PDSEL<1:0>		STSEL
bit 7							bit 0

图注：	HC = 用硬件清零		
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15　**UARTEN**：UARTx 使能位
1 = 使能 UARTx； UARTx 根据 UEN<1:0> 的定义控制所有 UARTx 引脚
0 = 禁止 UARTx；由端口锁存器控制所有 UARTx 引脚； UARTx 的功耗最小

bit 14　**未实现**：读为 0

bit 13　**USIDL**：空闲模式停止位
1 = 当器件进入空闲模式时，模块停止工作
0 = 在空闲模式下模块继续工作

bit 12　**IREN**：IrDA 编码器和解码器使能位 (1)
1 = 使能 IrDA® 编码器和解码器
0 = 禁止 IrDA 编码器和解码器

bit 11　**RTSMD**：$\overline{\text{UxRTS}}$ 引脚模式选择位
1 = $\overline{\text{UxRTS}}$ 引脚处于单工模式
0 = UxRTS 引脚处于流控制模式

bit 10　**未实现**：读为 0

bit 9-8　**UEN<1:0>**：UARTx 使能位
11 = 使能并使用 UxTX、 UxRX 和 BCLK 引脚； $\overline{\text{UxCTS}}$ 引脚由端口锁存器控制
10 = 使能并使用 UxTX、 UxRX、 UxCTS 和 UxRTS 引脚
01 = 使能并使用 UxTX、 UxRX 和 UxRTS 引脚； UxCTS 引脚由端口锁存器控制
00 = 使能并使用 UxTX 和 UxRX 引脚； UxCTS 和 UxRTS/BCLK 引脚由端口锁存器控制

bit 7　**WAKE**：在休眠模式下检测到启动位唤醒使能位
1 = UARTx 将继续采样 UxRX 引脚；在出现下降沿时产生中断；在出现上升沿时由硬件清零该位
0 = 禁止唤醒

bit 6　**LPBACK**：UARTx 环回模式选择位
1 = 使能环回模式
0 = 禁止环回模式

bit 5　**ABAUD**：自动波特率使能位
1 = 使能对下一个字符的波特率测量——需要在接收其他数据前接收同步字段 （55h）；完成时由硬件清零
0 = 禁止波特率检测或检测已完成

bit 4　**URXINV**：接收极性翻转位
1 = UxRX 的空闲状态为 0
0 = UxRX 的空闲状态为 1

注　1：此功能只能在 16 倍频 BRG 模式 （BRGH = 0）下使用。
2：位的可用性取决于引脚的可用性。

© 2007 Microchip Technology Inc.

寄存器 17-1：　UxMODE：UARTx 模式寄存器 （续）

bit 3　**BRGH：**高波特率使能位
1 = BRG 在每个位周期内产生 4 个时钟信号 （4 倍频波特率时钟，高速模式）
0 = BRG 在每个位周期内产生 16 个时钟信号 （16 倍频波特率时钟，标准模式）

bit 2-1　**PDSEL<1:0>：**奇偶校验和数据选择位
11 = 9 位数据，无奇偶校验
10 = 8 位数据，奇校验
01 = 8 位数据，偶校验
00 = 8 位数据，无奇偶校验

bit 0　**STSEL：**停止位选择位
1 = 2 个停止位
0 = 1 个停止位

注　1： 此功能只能在 16 倍频 BRG 模式 （BRGH = 0）下使用。
2： 位的可用性取决于引脚的可用性。

寄存器 17-2： UxSTA：UARTx 状态和控制寄存器

R/W-0	R/W-0	R/W-0	U-0	R/W-0 HC	R/W-0	R-0	R-1
UTXISEL1	UTXINV(1)	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R-1	R-0	R-0	R/C-0	R-0
URXISEL<1:0>		ADDEN	RIDLE	PERR	FERR	OERR	URXDA
bit 7							bit 0

图注：	HC = 用硬件清零		
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15,13 **UTXISEL<1:0>：**发送中断模式选择位
11 = 保留；不要使用
10 = 当一个字符被传输到发送移位寄存器导致发送缓冲器为空时，产生中断
01 = 当最后一个字符被移出发送移位寄存器；所有发送操作执行完毕时产生中断
00 = 当一个字符被传输到发送移位寄存器 （这意味着发送缓冲器中至少还有一个字符）时产生中断

bit 14 **UTXINV：**IrDA 编码器发送极性翻转位 (1)
1 = IrDA® 编码的 UxTX 空闲状态为 1
0 = IrDA 编码的 UxTX 空闲状态为 0

bit 12 **未实现：**读为 0

bit 11 **UTXBRK：**发送间隔位
1 = 在下次发送时发出同步间隔字符——启动位，后跟 12 个 0 位，然后是停止位；完成时由硬件清零
0 = 禁止或已完成同步间隔字符的发送

bit 10 **UTXEN：**发送使能位
1 = 使能发送， UARTx 控制 UxTX 引脚
0 = 禁止发送，中止所有等待的发送，缓冲器复位。由端口控制 UxTX 引脚。

bit 9 **UTXBF：**发送缓冲器满状态位 （只读）
1 = 发送缓冲器满
0 = 发送缓冲器未满，至少还可写入一个或多个字符

bit 8 **TRMT：**发送移位寄存器空位 （只读）
1 = 发送移位寄存器为空，同时发送缓冲器为空 （上一次发送已完成）
0 = 发送移位寄存器非空，发送在进行中或在发送缓冲器中排队

bit 7-6 **URXISEL<1:0>：**接收中断模式选择位
11 = 当 UxRSR 传输使接收缓冲器为满时 （即，有 4 个数据字符），中断标志位置 1
10 = 当 UxRSR 传输使接收缓冲器 3/4 满时 （即，有 3 个数据字符），中断标志位置 1
0x = 当接收到一个字符时，中断标志位置 1；且 UxRSR 的内容被传输给接收缓冲器。接收缓冲器有一个或多个字符。

bit 5 **ADDEN：**地址字符检测位 （接收数据的第 8 位 = 1）
1 = 使能地址检测模式。如果没有选择 9 位模式，这个控制位将无效。
0 = 禁止地址检测模式

注 1： 仅当使能了 IrDA 编码器 （IREN = 1）时，该位的值才影响模块的发送属性。

© 2007 Microchip Technology Inc.

寄存器 17-2：　　UxSTA：UARTx 状态和控制寄存器 （续）

bit 4　　**RIDLE：**接收器空闲位 （只读）
1 = 接收器空闲
0 = 接收器工作

bit 3　　**PERR：**奇偶校验错误状态位 （只读）
1 = 检测到当前字符 （在接收 FIFO 顶部的字符）的奇偶校验错误
0 = 未检测到奇偶校验错误

bit 2　　**FERR：**帧错误状态位 （只读）
1 = 检测到当前字符 （在接收 FIFO 顶部的字符）的帧错误
0 = 未检测到帧错误

bit 1　　**OERR：**接收缓冲器溢出错误状态位 （只读 / 清零）
1 = 接收缓冲器已溢出
0 = 接收缓冲器未溢出。清零原来置1的OERR位（1 → 0的跳变）将使接收缓冲器复位并使UxRSR为空。

bit 0　　**URXDA：**接收缓冲器中是否有数据位 （只读）
1 = 接收缓冲器中有数据，有至少一个或多个字符可被读取
0 = 接收缓冲器为空

注　1： 仅当使能了 IrDA 编码器 （IREN = 1）时，该位的值才影响模块的发送属性。

注：

© 2007 Microchip Technology Inc.

18.0 增强型 CAN （ECAN™）模块

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10 器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

18.1 概述

增强型控制器局域网（Enhanced Controller Area Network，ECAN）模块是一个串行接口，用于同其他 CAN 模块或单片机器件进行通信。此接口 / 协议是针对允许在噪声环境下通信而设计的。dsPIC33FJXXXGPX06/X08/X10 器件最多带有 2 个 ECAN 模块。

CAN 模块是一个通信控制器，实现了 BOSCH 规范中定义的 CAN 2.0 A/B 协议。该模块将支持 CAN 1.2、CAN 2.0A、CAN 2.0B Passive 和 CAN 2.0B Active 版本的协议。该模块实现了一种完整的 CAN 系统。但是本数据手册不讨论CAN规范。更多详细信息，读者可参见BOSCH CAN 规范。

该模块具有以下特性：

- 实现了CAN协议CAN 1.2、CAN 2.0A和CAN 2.0B
- 支持标准和扩展数据帧
- 0-8 字节数据长度
- 高达 1 Mb/s 的可编程比特率
- 自动响应远程发送请求
- 最多 8 个发送缓冲器，可由应用程序指定优先级和中止功能 （每个缓冲器最多包含 8 个字节的数据）
- 最多 32 个接收缓冲器（每个缓冲器最多包含 8 个字节的数据）
- 最多 16 个完全（标准 / 扩展标识符）的接收过滤器
- 3 个完全接收过滤屏蔽寄存器
- 支持 DeviceNet™ 寻址
- 集成了低通滤波器的可编程唤醒功能
- 支持自检操作的可编程环回模式
- 通过中断功能在出现所有的 CAN 接收器和发送器错误条件时发出信号
- 可编程时钟源
- 与输入捕捉模块的可编程连接（CAN1 和 CAN2 的 IC2），以进行时间标记和网络同步
- 低功耗休眠和空闲模式

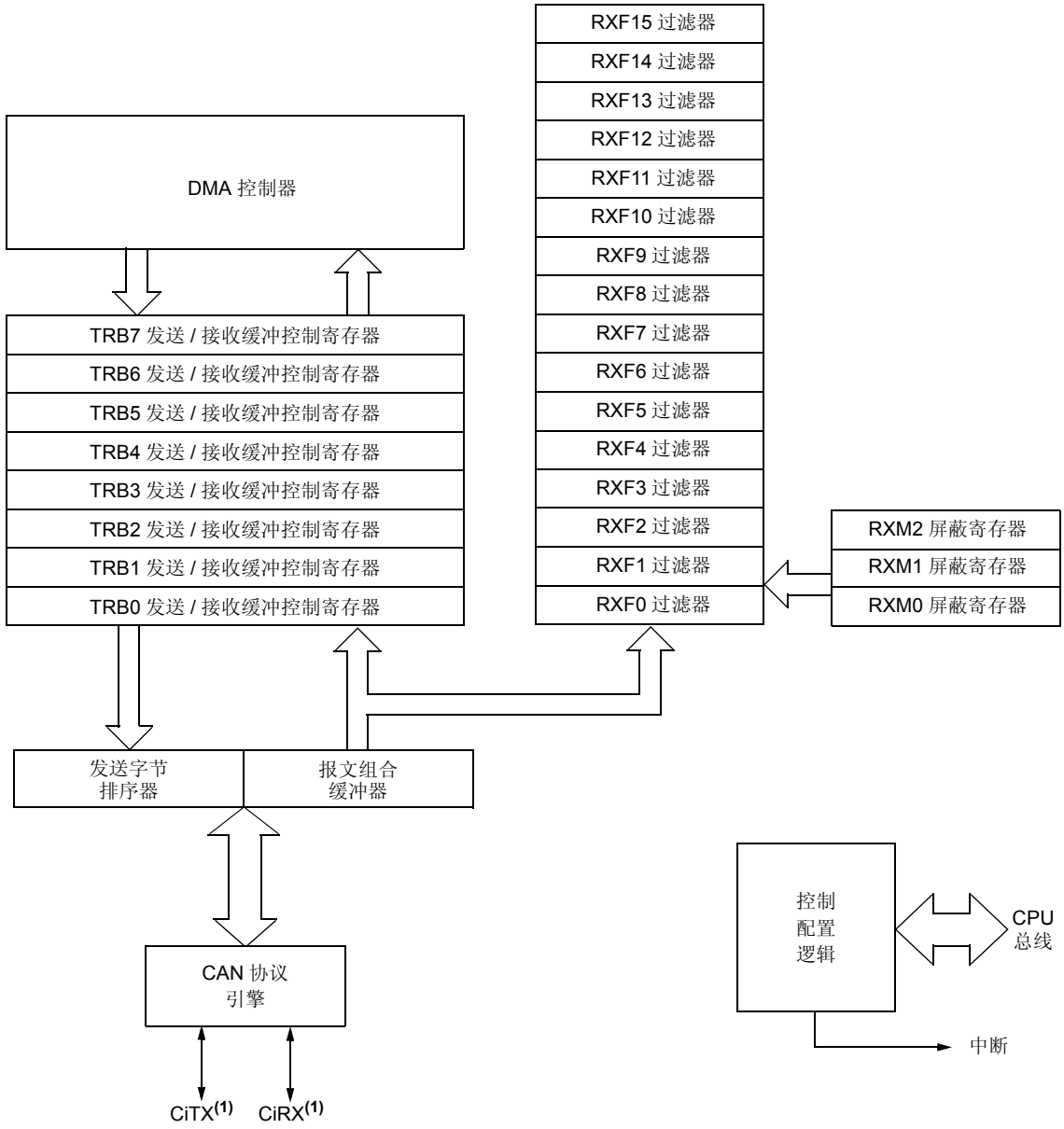
CAN 总线模块由协议引擎与报文缓冲 / 控制模块组成。CAN 协议引擎处理在 CAN 总线上接收和发送报文的所有功能。通过首先装载相应的数据寄存器发送报文。通过读取相应的寄存器可以检测状态和错误。将对在 CAN 总线上检测到的任何报文进行错误检测，并随后将其与过滤器进行比较以判断是否要将其接收并存储到接收寄存器之一中。

18.2 帧类型

CAN 模块支持包括数据报文或由用户发出的远程发送请求在内的各种类型的帧，以及自动产生用于控制的其他帧。支持下列帧类型：

- 标准数据帧：
  当节点要发送数据时会产生一个标准数据帧。它包含一个 11 位的标准标识符 （Standard Identifier，SID），而不是 18 位的扩展标识符 （Extended Identifier， EID）。
- 扩展数据帧：
  扩展数据帧与标准数据帧相似，但包含一个扩展标识符。
- 远程帧：
  也可能发生目标节点向源节点请求发送数据的情况。要做到这一点，目标节点必须发送一个其标识符与所需的数据帧匹配的远程帧。随后相应的数据源节点会发送一个数据帧作为对这个远程请求的响应。
- 错误帧：
  错误帧是由检测到总线错误的任一节点产生的。错误帧包含 2 个字段：错误标志字段和错误界定符字段。
- 过载帧：
  节点在两种条件下会产生过载帧：第一，节点在帧间间隔内检测到一个显性位，这是一种非法的情况。第二，由于内部原因，节点尚无法开始接收下一条报文。节点最多可产生 2 个连续过载帧来延迟下一条报文的接收。
- 帧间间隔：
  帧间间隔将前一个帧（无论何种类型）与其后的数据帧或远程帧分隔开来。

图 18-1：　　ECAN™ 模块框图



注 1： i = 1 或 2 指某个 ECAN 模块（ECAN1 或 ECAN2）。

© 2007 Microchip Technology Inc.

18.3 工作模式

用户可以选择 CAN 模块在以下几种工作模式之一工作。这些模式包括：

- 初始化模式
- 禁止模式
- 正常工作模式
- 监听模式
- 监听所有报文模式
- 环回模式

通过设置 REQOP<2:0> 位（CiCTRL1<10:8>）可选择所需模式。通过监视 OPMODE<2:0> 位（CiCTRL1<7:5>）可以确定进入的模式。通常在总线上检测到至少 11 个连续隐性位表明总线空闲时才允许改变模式，在此之前不会改变模块的工作模式和 OPMODE 位。

18.3.1 初始化模式

在初始化模式下，模块将不进行发送或接收。错误计数器被清零且中断标志位保持不变。编程人员可以访问在其他模式下不可访问的配置寄存器。模块会防止用户因为编程错误而意外违反 CAN 协议。当模块在线时，所有控制模块配置的寄存器都不能被修改。当进行发送的时候，不允许 CAN 模块进入配置模式。配置模式会作为锁来保护以下寄存器：

- 所有模块的控制寄存器
- 波特率和中断配置寄存器
- 总线时序寄存器
- 标识符接收过滤寄存器
- 标识符接收屏蔽寄存器

18.3.2 禁止模式

在禁止模式下，模块不会进行发送或接收。由于总线活动，模块能够将 WAKIF 位置 1，但是，等待处理的中断将继续等待，且错误计数器的值也将保持不变。

如果 REQOP<2:0> 位（CiCTRL1<10:8>）= 001，模块将进入模块禁止模式。如果模块处于工作状态，它将等待 CAN 总线上出现 11 个隐性位，表明总线空闲，然后才能执行模块禁止命令。当 OPMODE<2:0> 位（CiCTRL1<7:5>）= 001 时，表明模块成功进入了模块禁止模式。当模块处于模块禁止模式时，I/O 引脚将恢复为普通 I/O 功能。

当模块或 CPU 处于休眠模式时，通过对模块进行编程可以在CiRX输入线中应用低通滤波器功能。由WAKFIL 位 （CiCFG2<14>）使能或禁止该滤波器。

注： 通常，如果允许 CAN 模块在某种工作模式下发送，并且在 CAN 模块进入该模式后立即被要求发送，则模块将在开始发送前等待总线上出现 11 个连续隐性位。如果用户在此 11 个隐性位期间切换到禁止模式，则发送会被中止，同时相应的TXABT位置1，TXREQ 位清零。

18.3.3 正常工作模式

当 REQOP<2:0> = 000 时选择正常工作模式。在该模式下，模块被激活，I/O 引脚将承担 CAN 总线功能。模块将通过CiTX和CiRX引脚发送和接收CAN总线报文。

18.3.4 监听模式

如果激活监听模式，则 CAN 总线上的模块处于被动状态。发送缓冲器恢复为端口 I/O 功能。接收引脚保持在输入状态。对于接收器，不发出错误标志或应答信号。该状态下，错误计数器不再工作。监听模式可用来检测 CAN 总线上的波特率。要使用它，必须有 2 个以上可以互相通信的节点。

18.3.5 监听所有报文模式

该模块能够设置为忽略所有错误并接收所有报文。通过将REQOP<2:0>设置为111可激活监听所有报文模式。在该模式下，报文组合缓冲器中的数据将被复制到接收缓冲器并可通过 CPU 接口读取，直到出现错误。

18.3.6 环回模式

如果激活环回模式，模块将在模块边界将内部发送信号连接到内部接收信号。发送和接收引脚将恢复为端口 I/O 功能。

18.4 报文接收

18.4.1 接收缓冲器

CAN总线模块最多有32个接收缓冲器，位于DMA RAM中。需要通过清零 CiTRmnCON 寄存器中相应的发送 / 接收缓冲器选择 （TXENn）位将前 8 个缓冲器配置为接收缓冲器。用户可以通过定义 DMABS<2:0> 位（CiFCTRL<15:13>）来选择 DMA RAM 中 CAN 缓冲区的总大小，前 16 个缓冲器可以分配给接收过滤器，其余的仅用作 FIFO 缓冲器。

另有一个缓冲器始终监视进入总线的报文。该缓冲器被称为报文组合缓冲器（Message Assembly Buffer，MAB）。

MAB 组合所有的输入报文。只有满足相应接收过滤器条件的报文才能被传输到相应的接收缓冲器中。当接收到报文时，RBIF 标志（CiINTF<1>）位置 1。这时用户需要检查 CiVEC 和 / 或 CiRXFUL1 寄存器来判定导致中断产生的过滤器和缓冲器。只有接收到报文时模块才将 RBIF 位置 1。用户处理完缓冲器中的报文后将该位清零。如果 RBIE 位置 1，在接收到报文时将产生中断。

18.4.2 FIFO 缓冲器模式

如果过滤器的缓冲器指针的值为 1111，那么 ECAN 模块提供 FIFO 缓冲功能。在该模式下，满足缓冲器要求的结果将写入 FIFO 中下一个可用的缓冲单元。

CiFCTRL 寄存器定义 FIFO 的大小。该寄存器中的 FSA<4:0> 位定义 FIFO 缓冲器的起始位置。如果使能 DMA，FIFO 的结束位置由 DMABS<2:0> 位定义。所以，FIFO 最多支持 32 个缓冲器。

18.4.3 报文接收过滤器

报文接收过滤器和屏蔽寄存器用于决定报文组合缓冲器中的报文是否应该被装入某个接收缓冲器中。一旦一条有效的报文被接收到报文组合缓冲器（MAB），报文的标识符字段就会与过滤值进行比较。如果匹配的话，该报文就会被装入相应的接收缓冲器。每个过滤器都与一个缓冲器指针（FnBP<3:0>）相关联，该指针用来将过滤器与 16 个接收缓冲器中的一个相关联。

接收过滤器通过检查进入报文中的 IDE 位（CiTRBnSID<0>）以决定如何比较标识符。如果 IDE 位清零，报文是标准帧，则只与 EXIDE 位（CiRXFnSID<3>）清零的过滤器比较。如果 IDE 位置 1，报文是扩展帧，则只与 EXIDE 位置 1 的过滤器比较。

18.4.4 报文接收过滤屏蔽寄存器

屏蔽位主要决定将对哪些位应用过滤器。如果任何屏蔽位被设置为零，无论过滤位为何值该位都会被自动接收。有 3 个与接收缓冲器相关的可编程接收过滤屏蔽寄存器。通过设置相应 CiFMSKSELn 寄存器中的 FnMSK<1:0> 位来选择所需的屏蔽寄存器，可以将这 3 个屏蔽寄存器中的任意一个与每个过滤器关联。

18.4.5 接收错误

CAN 模块将会检测到以下接收错误：

- 循环冗余校验（Cyclic Redundancy Check，CRC）错误
- 位填充错误
- 无效报文接收错误

这些接收错误不会产生中断。然而，当发生上述错误之一时，接收错误计数器会递增 1。RXWAR 位（CiINTF<9>）表明接收错误计数器已经达到 CPU 警告的上限值 96，并由此产生中断。

18.4.6 接收中断

接收中断主要分为 3 组，每组包含各种产生中断的条件：

- 接收中断：

  报文已被成功接收并被装入其中一个接收缓冲器。接收到帧结束（End-of-Frame，EOF）字段后立即激活中断。读 RXnIF 标志位可知哪个接收缓冲器引起了中断。

- 唤醒中断：

  CAN 模块从禁止模式中唤醒，或器件从休眠模式中唤醒。

- 接收错误中断：

  接收错误中断由 ERRIF 位表示。该位表示有错误条件发生。通过检查 CAN 中断标志寄存器 CiINTF 中的相应位，就可以确定错误源。

  - 收到无效报文：

    如果在接收上一个报文期间发生了任何类型的错误，IVRIF 位都将指出有错误发生。

  - 接收器溢出：

    RBOVIF 位（CiINTF<2>）表明有溢出情况发生。

  - 接收器警告：

    RXWAR 位表明接收错误计数器（RERRCNT<7:0>）已经达到 CPU 警告的上限值 96。

  - 接收器错误被动：

    RXEP 位表明接收错误计数器已经超过了错误被动的上限值 127，且该模块已经进入错误被动状态。

© 2007 Microchip Technology Inc.

18.5 报文发送

18.5.1 发送缓冲器

CAN 模块最多有 8 个发送缓冲器，位于 DMA RAM 中。需要通过将 CiTRmnCON 寄存器中相应的发送 / 接收缓冲器选择（TXENn 或 TXENm）位置 1，将这 8 个缓冲器配置为发送缓冲器。用户可以通过定义 DMABS<2:0> 位 （CiFCTRL<15:13>）来选择 DMA RAM 中 CAN 缓冲区的总大小。

每个发送缓冲器占用 16 个数据字节。其中的 8 个字节用于存放发送的报文（最大 8 个字节）。另外 5 个字节用来存放标准或扩展报文标识符和其他报文仲裁信息。最后一个字节不使用。

18.5.2 发送报文优先级

发送优先级指在各个节点内待发送报文的优先级。发送优先级有 4 级。如果一个指定报文缓冲器的 TXnPRI<1:0> 位（在 CiTRmnCON 中）被设置为 11，则该缓冲器具有最高优先级。如果一个指定报文缓冲器的 TXnPRI<1:0> 位被设置为 10 或 01，则该缓冲器具有中等优先级。如果一个指定报文缓冲器的 TXnPRI<1:0> 位设置为 00，则该缓冲器具有最低优先级。如果两个或更多等待发送的报文具有相同优先级，报文以缓冲器编号递减的顺序发送。

18.5.3 发送过程

必须将 TXREQn 位 （在 CiTRmnCON 中）置 1 开始发送报文。CAN 总线模块解决了置 1 TXREQn 位和帧起始（Start-of-Frame， SOF）之间的时序冲突，确保当优先级改变时，能在 SOF 产生之前正确解决时序冲突。当 TXREQn 位置 1 时，TXABTn、TXLARBn 和 TXERRn 标志位被自动清零。

将 TXREQn 位置 1 仅仅标志报文缓冲器正在排序等待发送。当模块检测到总线可用时，模块开始发送设定为具有最高优先级的报文。

如果第一次尝试就成功发送了报文，TXREQn 位将自动清零。如果 TXnIE 置 1，会产生一个中断。

如果报文发送失败，错误条件标志位之一将置 1，TXREQn 位将保持置 1，表示该报文仍然等待发送。如果报文在尝试发送过程中遇到错误，TXERRn 位将被置 1，错误条件会产生一个中断。如果报文在尝试发送过程中仲裁失败，TXLARBn 位将被置 1。在仲裁失败时不会产生中断。

18.5.4 远程发送请求的自动处理

如果指定发送缓冲器的 RTRENn 位 （在 CiTRmnCON 寄存器中）被置 1，硬件自动发送在该缓冲器中的数据，以响应与指向该特定缓冲器的过滤器相匹配的远程发送请求。在这种情况下，用户不需要手动开始发送。

18.5.5 中止报文发送

通过清零与各个报文缓冲器相关的 TXREQ 位，系统能中止报文发送。将 ABAT 位 （CiCTRL1<12>）置 1 将请求中止所有等待发送的报文。如果报文还未开始发送或者报文已开始发送但由于仲裁失败或错误而被中断，那么将会执行中止。当模块将 TXABT 位置 1 且 TXnIF 标志位未自动置 1 时，表明发生了中止。

18.5.6 发送错误

CAN 模块将会检测到以下发送错误：

- 应答错误
- 格式错误
- 位错误

这些发送错误不一定会产生中断，但是发送错误计数器中会显示有错误发生。并且，每个错误都会引起错误计数器的值递增 1。一旦错误计数器的值超过 96，ERRIF （CiINTF<5>）和 TXWAR 位 （CiINTF<10>）将被置 1。一旦错误计数器的值超过 96，就会产生中断，且中断标志寄存器中的 TXWAR 位被置 1。

18.5.7 发送中断

发送中断主要分为2组，每组包括各种产生中断的条件：

- 发送中断：
  三个发送缓冲器中至少有一个为空（未预定）并且可以装入按照预定时间发送的报文。读TXnIF标志位可知哪个发送缓冲器可用及哪个发送缓冲器引起了中断。
- 发送错误中断：
  发送错误中断由 ERRIF 标志位表示。该标志位表示有错误情况发生。通过检查 CAN 中断标志寄存器 CiINTF 中的错误标志，就可以确定错误源。该寄存器中的标志与接收和发送错误有关。
  - 发送器警告中断：
    TXWAR位表明发送错误计数器已经达到CPU警告的上限值 96。
  - 发送器错误被动：
    TXEP 位 （CiINTF<12>）表示发送错误计数器已经超过了错误被动的上限值 127，且该模块已经进入了错误被动状态。
  - 总线关闭：
    TXBO 位 （CiINTF<13>）表示发送错误计数器的值已经超过了 255，且该模块已经进入了总线关闭状态。

注：	ECAN1 和 ECAN2 都能触发 DMA 数据传输。如果 C1TX、C1RX、C2TX 或 C2RX 被选作 DMA IRQ 源，在由于 ECAN1 或 ECAN2 的发送或接收导致 C1TXIF、C1RXIF、 C2TXIF 或 C2RXIF 位置 1 时，会发生 DMA 传输。

18.6 波特率设置

任何特定 CAN 总线上的所有节点必须具有相同的标称比特率。为了设置波特率，必须对以下参数进行初始化：

- 同步跳转宽度
- 波特率预分频比
- 相位段
- 相位段 2 的长度确定
- 采样点
- 传播时间段位

18.6.1 位时序

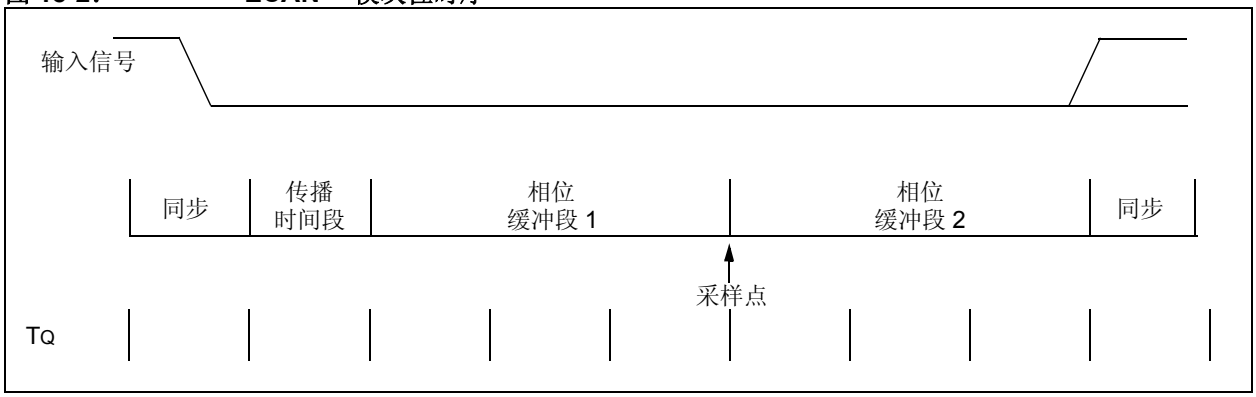
CAN 总线上的所有控制器必须使用相同的波特率和位长度。然而，并不要求所有的控制器具有相同的主振荡器时钟频率。由于各个控制器的时钟频率不同，必须通过调节各段的时间份额数来调整波特率。

可以认为，标称位时间可划分成几个互不重叠的时间段。这些段如图 18-2 所示。

- 同步段 （Sync Seg）
- 传播时间段 （Prop Seg）
- 相位缓冲段 1 （Phase1 Seg）
- 相位缓冲段 2 （Phase2 Seg）

时间段以及标称位时间由整数个时间单元组成，这些单元称作时间份额或 TQ。根据定义，标称位时间最少由 8 个 TQ 组成，最多由 25 个 TQ 组成。同样根据定义，最小标称位时间是 1 µs，对应最大 1 MHz 的比特率。

图 18-2： ECAN™ 模块位时序



© 2007 Microchip Technology Inc.

18.6.2 预分频比设置

模块带有一个可编程预分频器，其整数预分频值范围为 1 至 64，除对时钟进行固定的二分频还可提供其他时钟选项。时间份额（TQ）是源自振荡器周期的固定时间单元，如公式 18-1 所示。

注：	FCAN 不能超出 40 MHz。如果 CANCKS = 0，那么 FCY 不能超出 20 MHz。

公式 18-1： 时钟发生的时间份额

$$T_Q = 2\ (BRP<5:0> + 1)/F_{CAN}$$

18.6.3 传播时间段

这部分位时间用来补偿网络内的物理延时。这些延时包括总线线路上的信号传播时间以及节点的内部延时。通过设置 PRSEG<2:0> 位 （CiCFG2<2:0>），传播时间段长度可以编程为 1 TQ 到 8 TQ。

18.6.4 相位缓冲段

相位缓冲段用于将接收到位的采样点放置在发送位时间内的最佳位置。采样点在相位缓冲段 1 和相位缓冲段 2 之间。这些段可通过重新同步延长或缩短。相位缓冲段 1 的末尾决定一个位周期内的采样点。相位缓冲段 1 的持续时间可编程为 1 TQ 到 8 TQ。相位缓冲段 2 提供在发送下一数据前的延时，其持续时间也可被编程为 1 TQ 到 8 TQ 或是定义为与相位缓冲段 1 和信息处理时间（2 TQ）两者中的较大者相等。将 SEG1PH<2:0>（CiCFG2<5:3>）位置 1 初始化相位缓冲段 1，将 SEG2PH<2:0> （CiCFG2<10:8>）位置 1 初始化相位缓冲段 2。

在设置相位缓冲段长度时，必须符合下列要求：

传播时间段 + 相位缓冲段 1 ≥ 相位缓冲段 2

18.6.5 采样点

采样点是读总线电平并确定接收位的值的一个时间点。采样点发生在相位缓冲段 1 的末尾。若位时序较慢而且包含很多 TQ，可以在采样点对总线线路进行多次采样。由 CAN 总线确定接收位的值为采样到的三个值中出现次数最多的那个值。在采样点进行多次采样，且前两次采样的时间相隔 TQ/2。通过将 SAM 位（CiCFG2<6>）置 1 或清零，CAN 模块允许用户选择同一点采样三次或一次。

通常，位采样应该发生在位时间的 60-70% 左右，取决于系统参数。

18.6.6 同步

为了补偿总线上各节点间振荡频率的相移，每个CAN控制器必须能够与输入信号的相关信号沿同步。当检测到发送数据中的一个沿时，逻辑电路会将该沿的位置与预期时间 （同步段）比较。电路将随后调整相位缓冲段 1 和相位缓冲段 2 的值。有两种同步机制。

18.6.6.1 硬同步

硬同步仅当总线空闲期间有一个从隐性转变到显性的沿时，才被执行，它指示报文传输的开始。硬同步后，位时间计数器从同步段重新开始计数。硬同步强制引起硬同步的沿处于重新开始的位时间的同步段之内。如果产生硬同步，则在相应的位时间内不能再有重新同步。

18.6.6.2 重新同步

重新同步可能使相位缓冲段 1 延长或使相位缓冲段 2 缩短。相位缓冲段延长或缩短量的上限由同步跳转宽度给出 （由 SJW<1:0> 位 （CiCFG1<7:6>）指定）。同步跳转宽度值将加到相位缓冲段 1 或从相位缓冲段 2 中减去。重新同步跳转宽度可以编程为 1 TQ 到 4 TQ。

在设置 SJW<1:0> 位时，必须符合下列要求：

相位缓冲段 2 > 同步跳转宽度

注：	在下面的寄存器说明中，寄存器标识符中的 “i” 表示特定的 ECAN 模块 （ECAN1 或 ECAN2）。 寄存器标识符中的 “n” 表示缓冲器、过滤器或屏蔽寄存器的编号。 寄存器标识符中的 “m” 表示特定 CAN 数据字段中的字数。

寄存器 18-1：　CiCTRL1：ECAN 控制寄存器 1

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0
—	—	CSIDL	ABAT	CANCKS	REQOP<2:0>		
bit 15							bit 8

R-1	R-0	R-0	U-0	R/W-0	U-0	U-0	R/W-0
OPMODE<2:0>			—	CANCAP	—	—	WIN
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-14　**未实现：**读为 0

bit 13　**CSIDL：**空闲模式停止位
1 = 当器件进入空闲模式时，模块停止工作
0 = 在空闲模式下模块继续工作

bit 12　**ABAT：**中止所有等待处理的发送位
通知所有发送缓冲器中止发送。模块将在所有发送中止时清零该位

bit 11　**CANCKS：**CAN 主时钟选择位
1 = CAN FCAN 时钟为 FCY
0 = CAN FCAN 时钟为 FOSC

bit 10-8　**REQOP<2:0>：**请求工作模式位
000 = 设置正常工作模式
001 = 设置禁止模式
010 = 设置环回模式
011 = 设置监听模式
100 = 设置配置模式
101 = 保留——不要使用
110 = 保留——不要使用
111 = 设置监听所有报文模式

bit 7-5　**OPMODE<2:0>：**工作模式位
000 = 模块工作在正常工作模式下
001 = 模块工作在禁止模式下
010 = 模块工作在环回模式下
011 = 模块工作在监听模式下
100 = 模块工作在配置模式下
101 = 保留
110 = 保留
111 = 模块工作在监听所有报文模式下

bit 4　**未实现：**读为 0

bit 3　**CANCAP：**CAN 报文接收定时器捕捉事件使能位
1 = 使能基于 CAN 报文接收的输入捕捉
0 = 禁止 CAN 捕捉

bit 2-1　**未实现：**读为 0

bit 0　**WIN：**SFR 映射窗口选择位
1 = 使用过滤器窗口
0 = 使用缓冲器窗口

© 2007 Microchip Technology Inc.

寄存器 18-2：　　CiCTRL2：ECAN 控制寄存器 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R-0	R-0	R-0	R-0	R-0
—	—	—	DNCNT<4:0>				
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-5　**未实现：**读为 0

bit 4-0　**DNCNT<4:0>**：DeviceNet™ 过滤器位编号位
10010-11111 = 无效选择
10001 = 最多可将数据字节 3 的 bit 6 与 EID<17> 做比较
.
.
.
00001 = 最多可将数据字节 1 的 bit 7 与 EID<0> 做比较
00000 = 不比较数据字节

寄存器 18-3：　CiVEC：ECAN 中断编码寄存器

U-0	U-0	U-0	R-0	R-0	R-0	R-0	R-0
—	—	—	FILHIT<4:0>				
bit 15							bit 8

U-0	R-1	R-0	R-0	R-0	R-0	R-0	R-0
—	ICODE<6:0>						
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-13　**未实现：**读为 0

bit 12-8　**FILHIT<4:0>：**选中的过滤器编号位
10000-11111 = 保留
01111 = 过滤器 15
.
.
.
00001 = 过滤器 1
00000 = 过滤器 0

bit 7　**未实现：**读为 0

bit 6-0　**ICODE<6:0>：**中断标志编码位
1000101-1111111 = 保留
1000100 = FIFO 几乎满中断
1000011 = 接收器溢出中断
1000010 = 唤醒中断
1000001 = 错误中断
1000000 = 无中断
0010000-0111111 = 保留
0001111 = RB15 缓冲器中断
.
.
.
0001001 = RB9 缓冲器中断
0001000 = RB8 缓冲器中断
0000111 = TRB7 缓冲器中断
0000110 = TRB6 缓冲器中断
0000101 = TRB5 缓冲器中断
0000100 = TRB4 缓冲器中断
0000011 = TRB3 缓冲器中断
0000010 = TRB2 缓冲器中断
0000001 = TRB1 缓冲器中断
0000000 = TRB0 缓冲器中断

© 2007 Microchip Technology Inc.

寄存器 18-4：　CiFCTRL：ECAN FIFO 控制寄存器

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
DMABS<2:0>			—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	FSA<4:0>				
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-13　**DMABS<2:0>**：DMA 缓冲器大小位
111 = 保留
110 = DMA RAM 中的 32 个缓冲器
101 = DMA RAM 中的 24 个缓冲器
100 = DMA RAM 中的 16 个缓冲器
011 = DMA RAM 中的 12 个缓冲器
010 = DMA RAM 中的 8 个缓冲器
001 = DMA RAM 中的 6 个缓冲器
000 = DMA RAM 中的 4 个缓冲器

bit 12-5　**未实现**：读为 0

bit 4-0　**FSA<4:0>**：FIFO 区域从哪个缓冲器开始位
11111 = RB31 缓冲器
11110 = RB30 缓冲器
.
.
.
00001 = TRB1 缓冲器
00000 = TRB0 缓冲器

寄存器 18-5：　CiFIFO：ECAN FIFO 状态寄存器

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	FBP<5:0>					
bit 15							bit 8

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	FNRB<5:0>					
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-14　**未实现：**读为 0

bit 13-8　**FBP<5:0>：**FIFO 写缓冲器指针位
011111 = RB31 缓冲器
011110 = RB30 缓冲器
.
.
.
000001 = TRB1 缓冲器
000000 = TRB0 缓冲器

bit 7-6　**未实现：**读为 0

bit 5-0　**FNRB<5:0>：**FIFO 下一个读缓冲器指针位
011111 = RB31 缓冲器
011110 = RB30 缓冲器
.
.
.
000001 = TRB1 缓冲器
000000 = TRB0 缓冲器

© 2007 Microchip Technology Inc.

寄存器 18-6：　CiINTF：ECAN 中断标志寄存器

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	TXBO	TXBP	RXBP	TXWAR	RXWAR	EWARN
bit 15							bit 8

R/C-0	R/C-0	R/C-0	U-0	R/C-0	R/C-0	R/C-0	R/C-0
IVRIF	WAKIF	ERRIF	—	FIFOIF	RBOVIF	RBIF	TBIF
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-14　**未实现：**读为 0

bit 13　**TXBO**：发送器处于错误状态总线关闭位

bit 12　**TXBP**：发送器处于错误状态总线被动位

bit 11　**RXBP**：接收器处于错误状态总线被动位

bit 10　**TXWAR**：发送器处于错误状态警告位

bit 9　**RXWAR**：接收器处于错误状态警告位

bit 8　**EWARN**：发送器或接收器处于错误状态警告位

bit 7　**IVRIF**：收到无效报文中断标志位

bit 6　**WAKIF**：总线唤醒活动中断标志位

bit 5　**ERRIF**：错误中断标志位 （CiINTF<13:8> 寄存器中的多个中断源）

bit 4　**未实现：**读为 0

bit 3　**FIFOIF**：FIFO 几乎满中断标志位

bit 2　**RBOVIF**：接收缓冲器溢出中断标志位

bit 1　**RBIF**：接收缓冲器中断标志位

bit 0　**TBIF**：发送缓冲器中断标志位

寄存器 18-7：　　CiINTE：ECAN 中断允许寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IVRIE	WAKIE	ERRIE	—	FIFOIE	RBOVIE	RBIE	TBIE
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-8　**未实现：**读为 0

bit 7　**IVRIE：**收到无效报文中断允许位

bit 6　**WAKIE：**总线唤醒活动中断允许位

bit 5　**ERRIE：**错误中断允许位

bit 4　**未实现：**读为 0

bit 3　**FIFOIE：**FIFO 几乎满中断允许位

bit 2　**RBOVIE：**接收缓冲器溢出中断允许位

bit 1　**RBIE：**接收缓冲器中断允许位

bit 0　**TBIE：**发送缓冲器中断允许位

© 2007 Microchip Technology Inc.

寄存器 18-8：　　CiEC：ECAN 发送 / 接收错误计数寄存器

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
TERRCNT<7:0>							
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
RERRCNT<7:0>							
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-8　**TERRCNT<7:0>**：发送错误计数位

bit 7-0　**RERRCNT<7:0>**：接收错误计数位

寄存器 18-9：　　CiCFG1：ECAN 波特率配置寄存器 1

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SJW<1:0>		BRP<5:0>					
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-8　**未实现：**读为 0

bit 7-6　**SJW<1:0>：**同步跳转宽度位
11 = 长度为 4 x TQ
10 = 长度为 3 x TQ
01 = 长度为 2 x TQ
00 = 长度为 1 x TQ

bit 5-0　**BRP<5:0>：**波特率预分频比位
11 1111 = TQ = 2 x 64 x 1/FCAN
00 0010 = TA = 2 x 3 x 1/FCAN
00 0001 = TA = 2 x 2 x 1/FCAN
00 0000 = TQ = 2 x 1 x 1/FCAN

© 2007 Microchip Technology Inc.

寄存器 18-10： CiCFG2：ECAN 波特率配置寄存器 2

U-0	R/W-x	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
—	WAKFIL	—	—	—	SEG2PH<2:0>		
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SEG2PHTS	SAM	SEG1PH<2:0>			PRSEG<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **未实现：**读为 0

bit 14 **WAKFIL：**选择是否使用 CAN 总线过滤器唤醒位
1 = 使用 CAN 总线过滤器唤醒
0 = 不使用 CAN 总线过滤器唤醒

bit 13-11 **未实现：**读为 0

bit 10-8 **SEG2PH<2:0>：**相位缓冲段 2 位
111 = 长度为 8 x TQ
000 = 长度为 1 x TQ

bit 7 **SEG2PHTS：**相位缓冲段 2 时间选择位
1 = 可自由编程
0 = SEG1PH 位的最大值或信息处理时间 （Information Processing Time， IPT）中的较大者

bit 6 **SAM：**CAN 总线采样位
1 = 总线在采样点被采样三次
0 = 总线在采样点被采样一次

bit 5-3 **SEG1PH<2:0>：**相位缓冲段 1 位
111 = 长度为 8 x TQ
000 = 长度为 1 x TQ

bit 2-0 **PRSEG<2:0>：**传播时间段位
111 = 长度为 8 x TQ
000 = 长度为 1 x TQ

寄存器 18-11： CiFEN1：ECAN 接收过滤器使能寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLTEN15	FLTEN14	FLTEN13	FLTEN12	FLTEN11	FLTEN10	FLTEN9	FLTEN8
bit 15							bit 8

R/W-0	R/W-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
FLTEN7	FLTEN6	FLTEN5	FLTEN4	FLTEN3	FLTEN2	FLTEN1	FLTEN0
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0 **FLTENn：** 使能过滤器 n 接收报文位
1 = 使能过滤器 n
0 = 禁止过滤器 n

寄存器 18-12： CiBUFPNT1：ECAN 过滤器 0-3 缓冲器指针寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F3BP<3:0>				F2BP<3:0>			
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F1BP<3:0>				F0BP<3:0>			
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-12 **F3BP<3:0>：** 当满足过滤器 3 的过滤条件时写接收缓冲器的位
bit 11-8 **F2BP<3:0>：** 当满足过滤器 2 的过滤条件时写接收缓冲器的位
bit 7-4 **F1BP<3:0>：** 当满足过滤器 1 的过滤条件时写接收缓冲器的位
bit 3-0 **F0BP<3:0>：** 当满足过滤器 0 的过滤条件时写接收缓冲器的位
1111 = 满足过滤条件的数据被接收到接收 FIFO 缓冲器中
1110 = 满足过滤条件的数据被接收到接收缓冲器 14 中
•
•
•
0001 = 满足过滤条件的数据被接收到接收缓冲器 1 中
0000 = 满足过滤条件的数据被接收到接收缓冲器 0 中

© 2007 Microchip Technology Inc.

寄存器 18-13： CiBUFPNT2：ECAN 过滤器 4-7 缓冲器指针寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F7BP<3:0>				F6BP<3:0>			
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F5BP<3:0>				F4BP<3:0>			
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-12 **F7BP<3:0>：**当满足过滤器 7 的过滤条件时写接收缓冲器的位

bit 11-8 **F6BP<3:0>：**当满足过滤器 6 的过滤条件时写接收缓冲器的位

bit 7-4 **F5BP<3:0>：**当满足过滤器 5 的过滤条件时写接收缓冲器的位

bit 3-0 **F4BP<3:0>：**当满足过滤器 4 的过滤条件时写接收缓冲器的位

寄存器 18-14： CiBUFPNT3：ECAN 过滤器 8-11 缓冲器指针寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F11BP<3:0>				F10BP<3:0>			
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F9BP<3:0>				F8BP<3:0>			
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-12 **F11BP<3:0>：**当满足过滤器 11 的过滤条件时写接收缓冲器的位

bit 11-8 **F10BP<3:0>：**当满足过滤器 10 的过滤条件时写接收缓冲器的位

bit 7-4 **F9BP<3:0>：**当满足过滤器 9 的过滤条件时写接收缓冲器的位

bit 3-0 **F8BP<3:0>：**当满足过滤器 8 的过滤条件时写接收缓冲器的位

寄存器 18-15：　CiBUFPNT4：ECAN 过滤器 12-15 缓冲器指针寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F15BP<3:0>				F14BP<3:0>			
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F13BP<3:0>				F12BP<3:0>			
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-12　**F15BP<3:0>：**当满足过滤器 15 的过滤条件时写接收缓冲器的位

bit 11-8　**F14BP<3:0>：**当满足过滤器 14 的过滤条件时写接收缓冲器的位

bit 7-4　**F13BP<3:0>：**当满足过滤器 13 的过滤条件时写接收缓冲器的位

bit 3-0　**F12BP<3:0>：**当满足过滤器 12 的过滤条件时写接收缓冲器的位

© 2007 Microchip Technology Inc.

寄存器 18-16： CiRXFnSID：ECAN 接收过滤器 n 标准标识符 （n = 0, 1, ..., 15）

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
bit 15							bit 8

R/W-x	R/W-x	R/W-x	U-0	R/W-x	U-0	R/W-x	R/W-x
SID2	SID1	SID0	—	EXIDE	—	EID17	EID16
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-5 **SID<10:0>：**标准标识符位
1 = 报文地址位 SIDx 必须为 1 才能与过滤器匹配
0 = 报文地址位 SIDx 必须为 0 才能与过滤器匹配

bit 4 **未实现：**读为 0

bit 3 **EXIDE：** 扩展标识符使能位
如果 MIDE = 1：
1 = 只与带有扩展标识符地址的报文匹配
0 = 只与带有标准标识符地址的报文匹配
如果 MIDE = 0：
忽略 EXIDE 位。

bit 2 **未实现：**读为 0

bit 1-0 **EID<17:16>：**扩展标识符位
1 = 报文地址位 EIDx 必须为 1 才能与过滤器匹配
0 = 报文地址位 EIDx 必须为 0 才能与过滤器匹配

寄存器 18-17： CiRXFnEID：ECAN 接收过滤器 n 扩展标识符 （n = 0, 1, ..., 15）

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0 **EID<15:0>：**扩展标识符位
1 = 报文地址位 EIDx 必须为 1 才能与过滤器匹配
0 = 报文地址位 EIDx 必须为 0 才能与过滤器匹配

寄存器 18-18： CiFMSKSEL1：ECAN 过滤器 7-0 屏蔽选择寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F7MSK<1:0>		F6MSK<1:0>		F5MSK<1:0>		F4MSK<1:0>	
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F3MSK<1:0>		F2MSK<1:0>		F1MSK<1:0>		F0MSK<1:0>	
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-14 **F7MSK<1:0>：**过滤器 7 的屏蔽器源位

bit 13-12 **F6MSK<1:0>：**过滤器 6 的屏蔽器源位

bit 11-10 **F5MSK<1:0>：**过滤器 5 的屏蔽器源位

bit 9-8 **F4MSK<1:0>：**过滤器 4 的屏蔽器源位

bit 7-6 **F3MSK<1:0>：**过滤器 3 的屏蔽器源位

bit 5-4 **F2MSK<1:0>：**过滤器 2 的屏蔽器源位

bit 3-2 **F1MSK<1:0>：**过滤器 1 的屏蔽器源位

bit 1-0 **F0MSK<1:0>：**过滤器 0 的屏蔽器源位

11 = 无屏蔽

10 = 接收屏蔽寄存器 2 包含屏蔽值

01 = 接收屏蔽寄存器 1 包含屏蔽值

00 = 接收屏蔽寄存器 0 包含屏蔽值

© 2007 Microchip Technology Inc.

寄存器 18-19： CiRXMnSID：ECAN 接收过滤器屏蔽器 n 标准标识符

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
bit 15							bit 8

R/W-x	R/W-x	R/W-x	U-0	R/W-x	U-0	R/W-x	R/W-x
SID2	SID1	SID0	—	MIDE	—	EID17	EID16
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-5 **SID<10:0>**：标准标识符位
1 = 过滤器比较操作包含 SIDx 位
0 = 过滤器比较操作与 SIDx 位无关

bit 4 **未实现**：读为 0

bit 3 **MIDE**： 标识符接收模式位
1 = 只匹配与过滤器中 EXIDE 位对应的报文类型 （标准或扩展地址）
0 = 如果过滤器匹配则与标准或扩展地址报文匹配
（即，如果 （过滤器 SID） = （报文 SID）或 （过滤器 SID/EID） = （报文 SID/EID））

bit 2 **未实现**：读为 0

bit 1-0 **EID<17:16>**：扩展标识符位
1 = 过滤器比较操作包含 EIDx 位
0 = 过滤器比较操作与 EIDx 位无关

寄存器 18-20： CiRXMnEID：ECAN 接收过滤器屏蔽器 n 扩展标识符

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0 **EID<15:0>**：扩展标识符位
1 = 过滤器比较操作包含 EIDx 位
0 = 过滤器比较操作与 EIDx 位无关

寄存器 18-21： CiRXFUL1：ECAN 接收缓冲器满寄存器 1

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL15	RXFUL14	RXFUL13	RXFUL12	RXFUL11	RXFUL10	RXFUL9	RXFUL8
bit 15							bit 8

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL7	RXFUL6	RXFUL5	RXFUL4	RXFUL3	RXFUL2	RXFUL1	RXFUL0
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0 **RXFUL<15:0>**：接收缓冲器 n 满位
1 = 缓冲器为满 （由模块置 1）
0 = 缓冲器为空 （由应用软件清零）

寄存器 18-22： CiRXFUL2：ECAN 接收缓冲器满寄存器 2

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL31	RXFUL30	RXFUL29	RXFUL28	RXFUL27	RXFUL26	RXFUL25	RXFUL24
bit 15							bit 8

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL23	RXFUL22	RXFUL21	RXFUL20	RXFUL19	RXFUL18	RXFUL17	RXFUL16
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0 **RXFUL<31:16>**：接收缓冲器 n 满位
1 = 缓冲器为满 （由模块置 1）
0 = 缓冲器为空 （由应用软件清零）

© 2007 Microchip Technology Inc.

寄存器 18-23： CiRXOVF1：ECAN 接收缓冲器溢出寄存器 1

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF15	RXOVF14	RXOVF13	RXOVF12	RXOVF11	RXOVF10	RXOVF9	RXOVF8
bit 15							bit 8

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF7	RXOVF6	RXOVF5	RXOVF4	RXOVF3	RXOVF2	RXOVF1	RXOVF0
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0 **RXOVF<15:0>：**接收缓冲器 n 溢出位
1 = 模块对一个已满的缓冲器执行了写操作 （由模块置 1）
0 = 溢出条件被清除 （由应用软件清零）

寄存器 18-24： CiRXOVF2：ECAN 接收缓冲器溢出寄存器 2

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF31	RXOVF30	RXOVF29	RXOVF28	RXOVF27	RXOVF26	RXOVF25	RXOVF24
bit 15							bit 8

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF23	RXOVF22	RXOVF21	RXOVF20	RXOVF19	RXOVF18	RXOVF17	RXOVF16
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0 **RXOVF<31:16>：**接收缓冲器 n 溢出位
1 = 模块对一个已满的缓冲器执行了写操作 （由模块置 1）
0 = 溢出条件被清除 （由应用软件清零）

寄存器 18-25： CiTRmnCON：ECAN 发送 / 接收缓冲器 m 控制寄存器 （m = 0,2,4,6 ； n = 1,3,5,7）

R/W-0	R-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
TXENn	TXABTn	TXLARBn	TXERRn	TXREQn	RTRENn	TXnPRI<1:0>	
bit 15							bit 8

R/W-0	R-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
TXENm	TXABTm(1)	TXLARBm(1)	TXERRm(1)	TXREQm	RTRENm	TXmPRI<1:0>	
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-8 **请参见控制缓冲器 n 中 bit 7-0 的定义**

bit 7 **TXENm**：发送 / 接收缓冲器选择位
1 = 缓冲器 TRBn 是发送缓冲器
0 = 缓冲器 TRBn 是接收缓冲器

bit 6 **TXABTm**：报文中止位 (1)
1 = 中止报文
0 = 成功完成报文发送

bit 5 **TXLARBm**：报文仲裁失败位 (1)
1 = 报文在发送过程中仲裁失败
0 = 报文在发送过程中没有仲裁失败

bit 4 **TXERRm**：在发送过程中检测到错误位 (1)
1 = 报文发送时发生总线错误
0 = 报文发送时未发生总线错误

bit 3 **TXREQm**：报文发送请求位
将该位设置为 1 请求发送报文。当报文发送成功后，此位会自动清零。在该位置 1 的情况下清零该位（= 0）将请求中止报文。

bit 2 **RTRENm**：自动远程发送使能位
1 = 当接收到远程发送时，将 TXREQ 置 1
0 = 当接收到远程发送时， TXREQ 不受影响

bit 1-0 **TXmPRI<1:0>**：报文发送优先级位
11 = 最高报文优先级
10 = 中高报文优先级
01 = 中低报文优先级
00 = 最低报文优先级

注 1： 当 TXREQ 置 1 时可将此位清零。

© 2007 Microchip Technology Inc.

注： 缓冲器、 SID、 EID、 DLC、数据字段和接收状态寄存器位于 DMA RAM 中。

寄存器 18-26： CiTRBnSID：ECAN 缓冲器 n 标准标识符 （n = 0, 1, ..., 31）

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	SID10	SID9	SID8	SID7	SID6
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID5	SID4	SID3	SID2	SID1	SID0	SRR	IDE
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-13 **未实现：** 读为 0

bit 12-2 **SID<10:0>：** 标准标识符位

bit 1 **SRR：** 替代远程请求位
1 = 报文将请求远程发送
0 = 正常报文

bit 0 **IDE：** 扩展标识符位
1 = 报文将发送扩展标识符
0 = 报文将发送标准标识符

寄存器 18-27： CiTRBnEID：ECAN 缓冲器 n 扩展标识符 （n = 0, 1, ..., 31）

U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	—	EID17	EID16	EID15	EID14
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID13	EID12	EID11	EID10	EID9	EID8	EID7	EID6
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-12 **未实现：** 读为 0

bit 11-0 **EID<17:6>：** 扩展标识符位

寄存器 18-28： CiTRBnDLC：ECAN 缓冲器 n 数据长度控制 （n = 0, 1, ..., 31）

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID5	EID4	EID3	EID2	EID1	EID0	RTR	RB1
bit 15							bit 8

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	RB0	DLC3	DLC2	DLC1	DLC0
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-10 **EID<5:0>**：扩展标识符位

bit 9 **RTR**：远程发送请求位
1 = 报文将请求远程发送
0 = 正常报文

bit 8 **RB1**：保留的 bit 1
用户必须按 CAN 协议将该位设置为 0。

bit 7-5 **未实现**：读为 0

bit 4 **RB0**： 保留的 bit 0
用户必须按 CAN 协议将该位设置为 0。

bit 3-0 **DLC<3:0>**：数据长度编码位

寄存器 18-29： CiTRBnDm：ECAN 缓冲器 n 数据字段字节 m （n = 0, 1, ..., 31 ； m = 0, 1, ..., 7）(1)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
TRBnDm7	TRBnDm6	TRBnDm5	TRBnDm4	TRBnDm3	TRBnDm2	TRBnDm1	TRBnDm0
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 7-0 **TRnDm<7:0>**：数据字段缓冲器 n 字节 m 位

注 1： 最高字节包含缓冲器的 （m + 1）字节。

© 2007 Microchip Technology Inc.

寄存器 18-30： CiTRBnSTAT：ECAN 接收缓冲器 n 状态寄存器 （n = 0, 1, ..., 31）

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	FILHIT4	FILHIT3	FILHIT2	FILHIT1	FILHIT0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-13 **未实现：**读为 0

bit 12-8 **FILHIT<4:0>：**选中的过滤器编码位 （模块只能针对接收缓冲器执行写操作，不用于发送缓冲器）对导致写入此缓冲器的过滤器的编号进行编码。

bit 7-0 **未实现：**读为 0

注：

© 2007 Microchip Technology Inc.

19.0 数据转换器接口 （DCI）模块

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10 器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

19.1 模块简介

dsPIC33FJXXXGPX06/X08/X10 数据转换器接口（Data Converter Interface，DCI）模块允许与如下器件的简单接口：音频编码器 / 解码器 （编解码器）、 ADC 和 D/A 转换器。支持以下接口：

- 帧同步串行传输 （单通道或多通道）
- Inter-IC Sound （I^2S）接口
- AC-Link 兼容模式

DCI 模块提供下列通用特性：

- 可设置的字长度，最大为 16 位
- 支持最大 16 个时隙，帧最长为 256 位
- 为最多 4 次采样缓冲数据，无 CPU 开销

19.2 模块 I/O 引脚

有 4 个 I/O 引脚与 DCI 模块相关。当使能时，模块控制这 4 个引脚的数据方向。

19.2.1 CSCK 引脚

CSCK 引脚为 DCI 模块提供串行时钟。使用 DCICON1 SFR 中的 CSCKD 控制位，可以将 CSCK 引脚配置为输入或输出。当配置为输出时，由 dsPIC33FJXXXGPX06/X08/X10 提供串行时钟。当配置为输入时，则必须由外部器件提供串行时钟。

19.2.2 CSDO 引脚

当模块使能时，串行数据输出 （CSDO）引脚被配置为只能输出的引脚。只要发送数据，CSDO 引脚就会驱动串行总线。在不发送数据的 CSCK 周期内，CSDO 引脚根据CSDOM控制位的状态处于三态或驱动为0。这样，其他器件就能在 DCI 模块不使用的发送周期内，把数据放置到串行总线上去。

19.2.3 CSDI 引脚

当模块使能时，串行数据输入 （CSDI）引脚被配置为只能输入的引脚。

19.2.3.1 COFS 引脚

编解码器帧同步 （COFS）引脚用于同步发生在 CSDO 和 CSDI 引脚上的数据传输。COFS 引脚可以配置为输入或输出引脚。 DCICON1 寄存器中的 COFSD 控制位控制 COFS 引脚的数据方向。

当 CPU 在访问存储器映射的缓冲寄存器时， DCI 模块访问影子寄存器。

19.2.4 缓冲器数据对齐

数据值在缓冲器中总是以左对齐方式存储，因为大多数编解码器数据是用有符号二进制补码小数表示的。如果接收到的字长度小于 16 位，则接收缓冲寄存器中未使用的低位会被模块置为 0。如果发送字长小于 16 位，模块将忽略发送缓冲寄存器中未使用的低位。本文档的后续小节将说明字长的设置。

19.2.5 发送 / 接收移位寄存器

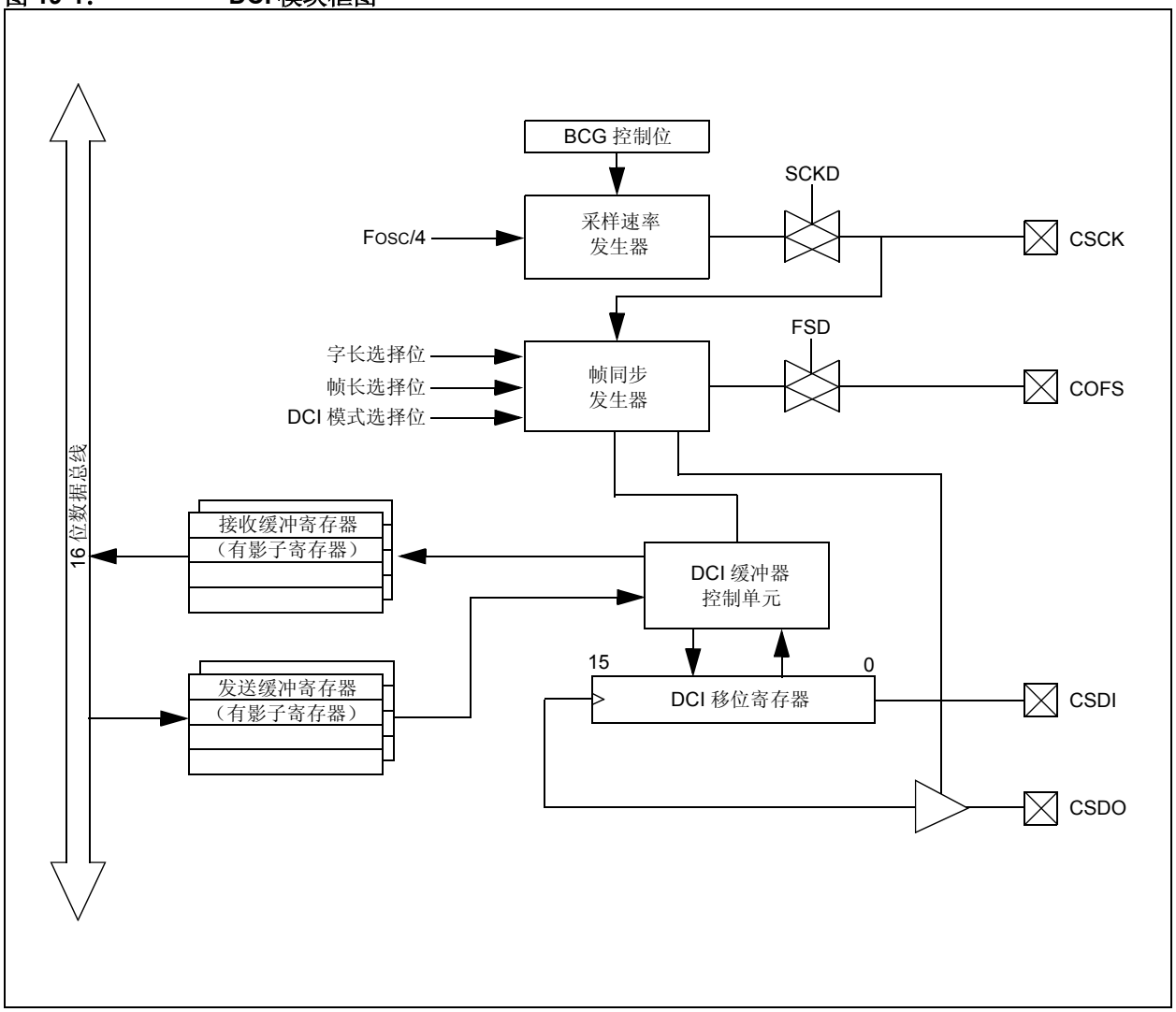
DCI 模块具备一个 16 位移位寄存器，用来把串行数据移入和移出模块。数据首先移入 / 移出移位寄存器的 MSb，因为音频 PCM 数据是以有符号二进制补码格式发送的。

19.2.6 DCI 缓冲器控制

DCI 模块包含一个缓冲器控制单元，用于在影子缓冲存储区和串行移位寄存器间传输数据。缓冲器控制单元是一个简单的 2 位地址计数器，指向影子缓冲存储器中的字存储单元。对于接收存储空间（DCI 缓冲存储区的高地址部分），地址计数器的值与一个为 0 的 MSb 位连在一起形成一个 3 位宽的地址。对于发送存储空间 （DCI 缓冲存储区的高地址部分），地址计数器的值与一个为 1 的 MSb 位相连。

注： DCI 缓冲器控制单元始终访问发送和接收缓冲器中相对地址相同的单元，所以模块只提供了一个地址计数器。

图 19-1：　　DCI 模块框图



© 2007 Microchip Technology Inc.

19.3 DCI 模块的工作

19.3.1 模块使能

通过将 DCICON1 SFR 中的 DCIEN 控制位置 1 或清零可以使能或禁止 DCI 模块。将 DCIEN 控制位清零将复位模块。尤其是，与 CSCK 发生、帧同步和 DCI 缓冲器控制单元相关的所有计数器都将复位。

当 DCIEN 位清零时， DCI 时钟关闭。

当 DCI 使能时，它控制与模块相关的 4 个 I/O 引脚的数据方向。当 DCIEN 位置 1 时，这些 I/O 引脚的 PORT、LAT 和 TRIS 寄存器的值都会被 DCI 模块改写。

当使能了位时钟发生器时，单独改写 CSCK 引脚的上述寄存器值也是允许的。这可以无需使能 DCI 模块的其余部分，就能让位时钟发生器工作。

19.3.2 字长选择位

DCICON2 SFR 中的 WS<3:0> 字长选择位决定每个 DCI 数据字的位数。本质上，WS<3:0> 位决定 4 位计数器的计数周期 （计数器时钟来自 CSCK 信号）。

可以选择任何长度的数据字长，字长最大 16 位。装入到 WS<3:0> 位中的值，要比期望的字长小 1。例如，如果 WS<3:0> = 1111，则选择 16 位数据字长。

注：	WS<3:0> 控制位仅用在多通道和 I^2S 模式中。这些位对 AC-Link 模式没有影响，因为该协议规定数据间隙的大小固定不变。

19.3.3 帧同步发生器

帧同步发生器 （COFSG）是一个 4 位计数器，它设置数据字中的帧长度。每次字长计数器复位时，帧同步发生器就递增一次（见**第 19.3.2 节 “字长选择位”**）。通过写 DCICON2 SFR 中的 COFSG<3:0> 控制位，来设置帧同步发生器的周期。以时钟周期表示的 COFSG 周期由以下公式决定：

公式 19-1： COFSG 周期

帧长 = 字长 • (FSG 值 + 1)

可以选择的帧长度最大为 16 个数据字。以 CSCK 周期表示的帧长度会随选择的字长而变化，最大可以为 256。

注：	COFSG 控制位在 AC-Link 模式中不起作用，因为该协议已经将帧长度设置为 256 个 CSCK 周期。

19.3.4 帧同步模式控制位

使用 DCICON1 SFR 中的帧同步模式控制位（COFSM<1:0>），来选择帧同步信号的类型。可选择以下工作模式：

- 多通道模式
- I^2S 模式
- AC-Link 模式 （16 位）
- AC-Link 模式 （20 位）

COFSM 控制位的操作，取决于 DCI 模块是作为主器件产生帧同步信号，还是作为从器件接收帧同步信号。

DSP/ 编解码器对中的主器件是产生帧同步信号的器件。帧同步信号启动 CSDI 和 CSDO 引脚上的数据传输，通常与数据采样速率 （COFS）具有相同的频率。

如果 COFSD 控制位清零，则 DCI 模块是帧同步主器件；如果 COFSD 控制位置 1，则 DCI 模块是帧同步从器件。

19.3.5 主器件帧同步操作

当 DCI 模块作为帧同步主器件（COFSD = 0）工作时，COFSM 模式位决定帧同步脉冲的类型；帧同步脉冲由帧同步发生器逻辑产生。

当帧同步发生器复位至 0 时，将产生新的 COFS 信号。

在多通道模式下，帧同步脉冲被驱动为高电平一个 CSCK 周期，从而启动数据传输。相继帧同步脉冲之间的 CSCK 周期数，取决于字长和帧同步发生器控制位。多通道模式下帧同步信号的时序图如图 19-2 所示。

在 AC-Link 工作模式下，帧同步信号有固定的周期和占空比。 AC-Link 帧同步信号维持 16 个 CSCK 周期的高电平和 240 个 CSCK 周期的低电平。图 19-3 给出了时序图和 AC-Link 帧起始的时序细节。

在 I^2S 模式下，产生占空比为 50% 的帧同步信号。 I^2S 帧同步信号的周期 （表示为多少 CSCK 周期）由字长和帧同步发生器控制位决定。COFS 引脚上电平从高至低或从低至高的跳变沿，标记出了新 I^2S 数据传输的边界。

19.3.6 从器件帧同步操作

当 DCI 模块作为帧同步从器件（COFSD = 1）工作时，数据传输由与 DCI 模块相连的编解码器控制。 COFSM 控制位控制 DCI 模块如何响应进入的 COFS 信号。

在多通道模式下，在 COFS 引脚采样到高电平一个 CSCK 周期后，将会开始新的数据帧传输（见图 19-2）。COFS 引脚上的脉冲会复位帧同步发生器逻辑。

在 I²S 模式下，在 COFS 引脚上采样到从低至高或从高至低的电平跳变一个 CSCK 周期后，会传输新的数据字。COFS 引脚的上升或下降沿将复位帧同步发生器逻辑。

在 AC-Link 模式下，当在 COFS 引脚采样到高电平一个 CSCK 周期后，将传输下一个帧的标记间隙和接下来的数据间隙。

当模块工作在从模式时，必须配置 COFSG 和 WS 位，提供恰当的帧长度。一旦模块在 COFS 引脚上采样到了有效的帧同步脉冲，就会发生一个完整的数据帧传输。在当前数据帧传输完成之前模块都不会响应其他帧同步脉冲。

图 19-2： 帧同步时序，多通道模式

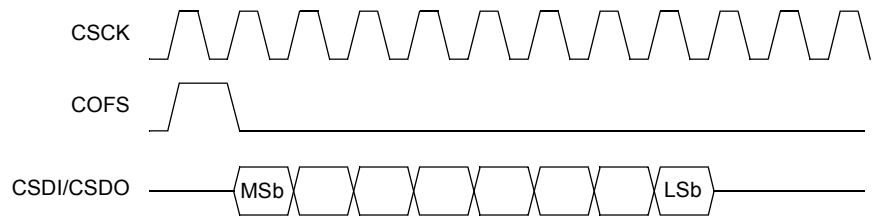


图 19-3： 帧同步时序， AC-LINK 帧起始

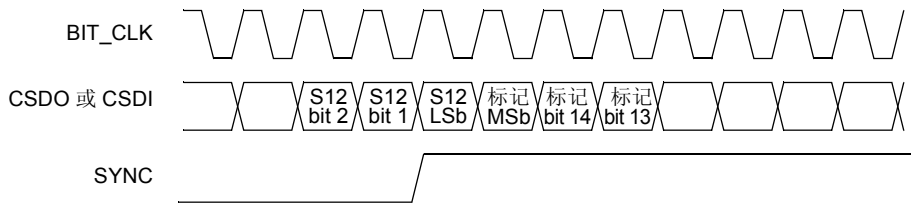
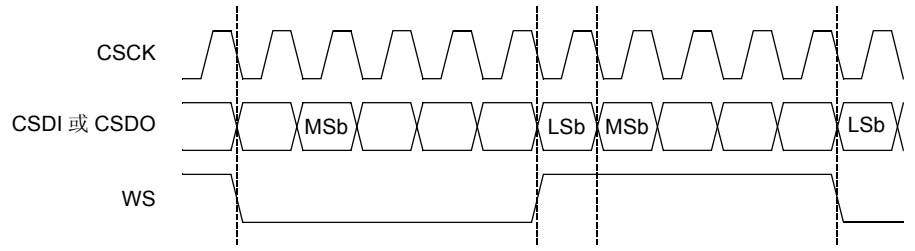


图 19-4： I²S 接口帧同步时序



注： 为便于说明，此处显示了 5 位传输。 I²S 协议没有规定字长，字长取决于系统。

© 2007 Microchip Technology Inc.

19.3.7 位时钟发生器

DCI 模块有一个专用的 12 位时基，用来产生位时钟。通过写一个非零的 12 位值到 DCICON3 SFR 中的 BCG<11:0> 控制位，设置位时钟的速率（周期）。

当 BCG<11:0> 位设置为零时，位时钟将禁止。如果把 BCG<11:0> 位设置为非零值，位时钟发生器被使能。如果 DCI 的串行时钟来自外部器件，这些位应该被设置为 0，且 CSCKD 位应设置为 1。

公式 19-2 给出了位时钟频率的计算公式。

公式 19-2： 位时钟频率

$$F_{BCK} = \frac{F_{CY}}{2 \bullet (BCG + 1)}$$

所需的位时钟频率将取决于系统采样速率和帧大小。根据数据转换器和使用的通信协议，典型的位时钟频率范围在 16 倍到 512 倍转换器采样速率之间。

为了获得与常见音频采样速率相关的位时钟频率，用户需要选择的晶振频率应该是“偶数”二进制值。表 19-1 列出了这类晶振频率的示例。

表 19-1： 用于常见编解码器 CSCK 频率的器件频率

FS （kHz）	FCSCK/FS	FCSCK （MHz）(1)	FOSC （MHz）	PLL	FCY （MIPS）	BCG(2)
8	256	2.048	8.192	4	8.192	1
12	256	3.072	6.144	8	12.288	1
32	32	1.024	8.192	8	16.384	7
44.1	32	1.4112	5.6448	8	11.2896	3
48	64	3.072	6.144	16	24.576	3

注 1： 当 CSCK 信号来自外部 （CSCKD = 1）时，外部时钟的高低电平时间必须满足器件时序要求。

2： 当 CSCK 信号来自外部 （CSCKD = 1）时， BCG<11:0> 位对 DCI 模块的工作不产生影响。

19.3.8 采样时钟边沿控制位

采样时钟边沿（CSCKE）控制位决定 CSCK 信号的采样边沿。如果 CSCK 位清零（默认），将在 CSCK 信号的下降沿采样数据。 AC-Link 协议和大部分多通道格式都要求在 CSCK 信号的下降沿采样数据。如果 CSCK 位置 1，将在 CSCK 的上升沿采样数据。 I^2S 协议要求在 CSCK 信号的上升沿采样数据。

19.3.9 数据对齐控制位

在大多数应用中，在 COFS 信号采样为高电平时，将在一个 CSCK 周期后开始数据传输。这是 DCI 模块的默认配置。通过将 DCICON1 SFR 中的 DJST 控制位置 1，可以选择另一种数据对齐方式。当 DJST = 1 时，如果 COFS 信号采样为高电平，将在当前 CSCK 周期中开始数据传输。

19.3.10 发送时隙使能位

TSCON SFR 具有的控制位可用于使能最多 16 个发送时隙。这些控制位是 TSE<15:0> 位。每个时隙的大小由 WS<3:0> 字长选择位决定，最大可以为 16 位。

如果通过某个 TSE 位（TSEx = 1）使能了发送时隙，当前发送影子缓冲单元中的内容就会被装入DCI 移位寄存器，同时 DCI 缓冲器控制单元将递增以指向下一个单元。

在未用的发送时隙中，CSDO 引脚将驱动为 0，或者在所有被禁止的时隙中呈现三态，这取决于 DCICON1 SFR 中 CSDOM 位的状态。

数据帧长度（按位计）由所选的数据字长、帧中数据字的个数决定。如果帧中数据字的个数少于 16，则多出的时隙使能位没有作用。

每个发送数据字以左对齐的方式写入 16 位发送缓冲器中。如果所选字长小于 16 位，发送缓冲存储区的低位对发送数据没有作用。用户应该向每个发送缓冲单元中未使用的低位写入 0。

19.3.11 接收时隙使能位

RSCON SFR 具有的控制位可用于使能最多 16 个接收时隙。这些控制位是 RSE<15:0> 位。每个接收时隙的大小由 WS<3:0> 字长选择位决定，大小可以从 1 位到 16 位。

如果通过某个 RSE 位（RSEx = 1）使能了接收时隙，DCI 移位寄存器的内容将被写入当前 DCI 接收影子缓冲单元中，同时缓冲器控制单元将递增以指向下一个缓冲单元。

如果所选的字长小于 16 位，数据在接收存储器缓冲单元中不被打包。每个接收时隙数据字都存储在一个单独的 16 位缓冲单元中。数据总是以左对齐的格式存储在接收存储缓冲器中。

19.3.12 帧同步时时隙使能位的操作

TSE 和 RSE 控制位与 DCI 帧同步发生器协同工作。在主模式下，一旦帧同步发生器复位，就将产生 COFS 信号。在从模式下，一旦接收到 COFS 脉冲，帧同步发生器就将复位。

TSE 和 RSE 控制位允许使能最多 16 个连续的发送或接收时隙。在最后一个使能的时隙发送 / 接收之后， DCI 将停止缓冲数据，直到下一次 COFS 脉冲产生。

19.3.13 同步数据传输

一旦使能了给定的发送或接收时隙，DCI 缓冲器控制单元将递增一个字单元。大多数情况下，数据输入和输出传输是同步的，这就是说，对于给定通道，数据采样的接收与发送是同时进行的。于是，当产生 DCI 中断时，发送和接收缓冲器中将装入相同数量的数据。

在一些情况下，一个数据帧内发送、接收的数据量可能不相等。例如，假设使用的是双字数据帧。此外，假设仅在 slot #0 中接收数据，但在 slot #0 和 slot #1 中发送数据。在这种情况下，在数据帧传输过程中，缓冲器控制单元计数器将递增两次，但只有一个接收寄存器单元将装入数据。

© 2007 Microchip Technology Inc.

19.3.14 缓冲器长度控制

两次中断之间缓冲的数据量，由 DCICON2 SFR 中的缓冲器长度（BLEN<1:0>）控制位决定。使用 BLEN 控制位，可以设置发送和接收缓冲器的长度为 1 至 4 个数据字。BLEN 控制位将与 DCI 缓冲器控制单元地址计数器的当前值进行比较。当 DCI 地址计数器的最低 2 位与 BLEN<1:0> 的值匹配时，缓冲器控制单元将复位为 0。此外，接收影子寄存器的内容将传送到接收缓冲寄存器中，发送缓冲寄存器的内容将传送到发送影子寄存器中。

注 1： DCI 能触发 DMA 数据传输。如果 DCI 被选作 DMA IRQ 源，在由于 DCI 发送或接收导致DCIIF位置1时，会发生DMA传输。

2： 如果需要 DMA 传输，必须将 DCI 发送 / 接收缓冲器的大小设置为 1 个字（即 BLEN<1:0> = 00）。

19.3.15 缓冲器与数据帧对齐

AGU 地址指针位置与数据帧边界之间没有直接的对应关系。这意味着，每个发送和接收缓冲器存在隐含的分配，这与 BLEN 控制位和使能的时隙数量（使能的时隙数量通过 TSE 和 RSE 控制位来设置）有关。

例如，假设选择了 4 字长的数据帧，我们打算在帧中的全部 4 个时隙都进行发送。通过设置 TSCON SFR 中的 TSE0、TSE1、TSE2 和 TSE3 控制位，可以得到这个配置。这样设置好模块后，TXBUF0 寄存器很自然地就会被分配给 slot #0，TXBUF1 寄存器很自然地就会被分配给 slot #1，依此类推。

注： 如果在一个数据帧中，有多于 4 个时隙有效，用户代码必须跟踪每次中断时读 / 写的时隙。在一些情况下，发送 / 接收缓冲器与对应时隙分配之间的对齐关系可能会丢失。仿真断点或硬件陷阱就是这样的例子。在这些情况下，用户应该查询 SLOT 状态位，决定在缓冲寄存器中装入什么数据，从而使软件与 DCI 模块重新同步。

19.3.16 发送状态位

在 DCISTAT SFR 中有两个发送状态位。

当发送缓冲寄存器内容被传送到发送影子寄存器中时，TMPTY 位置 1。可以用软件查询 TMPTY 位以确定发送缓冲寄存器被写入的时间。如果对 4 个发送缓冲器之一进行了写入，硬件将自动清零 TMPTY 位。

TUNF 位是只读的，它表明正在使用的发送缓冲寄存器中至少有一个发生了下溢。发送缓冲寄存器的内容被传送到发送影子寄存器时，TUNF 位就将置 1。当 CPU 对下溢的缓冲寄存器进行了写入时，将自动清零 TUNF 状态位。

注： 发送状态位仅用来指明模块所使用的缓冲单元的状态。例如，如果设置的缓冲器长度小于 4 个字，则未使用的缓冲单元将不会影响发送状态位。

19.3.17 接收状态位

在 DCISTAT SFR 中有两个接收状态位。

RFUL 状态位是只读的，它表明在接收缓冲器中是否有新数据。在 CPU 读取正在使用的全部接收缓冲器后，RFUL 位自动清零。

ROV状态位是只读的，它表明接收缓冲单元中至少有一个产生了接收溢出。如果在新数据从影子寄存器传送到缓冲单元之前，CPU 没有读取缓冲单元，就将产生接收溢出。如果 CPU 读取了造成溢出的缓冲寄存器， ROV 状态位将自动清零。

对于特定的缓冲单元，当发生接收溢出时，缓冲器中的原有内容将被改写。

注： 接收状态位仅用来指明模块所使用的缓冲单元的状态。例如，如果设置的缓冲器长度小于 4 个字，则未使用的缓冲单元将不会影响接收状态位。

19.3.18 SLOT 状态位

DCISTAT SFR 中的 SLOT<3:0> 状态位指明当前有效时隙。这些位与帧同步发生器的计数器值相对应。当产生 DCI 中断时，用户可以在软件中查询这些状态位，从而确定哪个时隙数据是最后接收到的以及哪个时隙数据应该被装入 TXBUF 寄存器。

19.3.19 CSDO 模式位

CSDOM 控制位控制 CSDO 引脚在未使用发送时隙中的状态。如果某个发送时隙在 TSCON SFR 中相应的 TSEx 位被清零，此发送时隙就是未使用的。

如果 CSDOM 位被清零（默认），在未使用时隙周期中，CSDO 引脚将为低电平。此模式用在仅有两个器件连接到串行总线时。

如果 CSDOM 位被置 1，CSDO 引脚在未使用的时隙周期中将呈现三态。在多通道应用中，此模式允许多个器件共用同一根 CSDO 线。此时应将 CSDO 线上的每个器件配置为只在特定时隙内发送数据。不允许有两个器件在同一时隙发送数据。

19.3.20 数字环回模式

通过将 DCICON1 SFR 中的 DLOOP 控制位置 1，使能数字环回模式。当 DLOOP 位置 1 时，模块将在内部将 CSDO 信号连入 CSDI 引脚。在数字环回模式下，将忽略 CSDI I/O 引脚上实际输入的数据。

19.3.21 下溢模式控制位

当发生下溢时，根据 DCICON1 SFR 中下溢模式（UNFM）控制位的状态，模块会执行两种操作之一。如果 UNFM 位清零（默认），在缓冲单元有效时隙期间，模块将在 CSDO 引脚上发送 0。在此工作模式下，连接到 DCI 模块的编解码器件将仅输入数字 “静音”。如果 UNFM 控制位置 1，模块将发送上一次写入缓冲单元的数据。该工作模式允许用户发送连续的数据到编解码器件，而无需 CPU 开销。

19.4 DCI 模块中断

DCI 模块中断的频率取决于 DCICON2 SFR 中的 BLEN<1:0> 控制位。每当达到设定的缓冲器长度且发生影子寄存器数据传送时，就会产生 CPU 中断。影子寄存器传送的定义是：上一次写入 TXBUF 的值被传送到发送影子寄存器，而接收影子寄存器中新接收的值被传送到 RXBUF 寄存器。

19.5 CPU 休眠和空闲模式期间 DCI 模块的工作

19.5.1 CPU 休眠模式期间 DCI 模块的工作

当 CSCK 信号由外部器件提供（CSCKD = 1）时，DCI 模块在 CPU 处于休眠模式时能够工作并唤醒 CPU。当 DCI 缓冲器传送完成且 CPU 处于休眠模式时， DCI 模块将产生一个异步中断。

19.5.2 CPU 空闲模式期间 DCI 模块的工作

如果 DCISIDL 控制位清零（默认），则模块将在空闲模式下继续正常工作。如果 DCISIDL 位置 1，当 CPU 进入空闲模式时，模块将停止工作。

19.6 AC-Link 模式操作

AC-Link 协议数据帧是 256 位的，它可以被分成一个 16 位的数据时隙和其后的 12 个 20 位的数据时隙。 DCI 模块有两种针对 AC-Link 协议的工作模式。使用 DCICON1 SFR 中的 COFSM<1:0> 控制位来选择这两种工作模式。第一种 AC-Link 模式称为 “16 位 AC-Link 模式”，通过设置 COFSM<1:0> = 10 来选择。第二种 AC-Link 模式称为 “20 位 AC-Link 模式”，通过设置 COFSM<1:0> = 11 来选择。

19.6.1 16 位 AC-LINK 模式

在 16 位 AC-Link 模式下，数据字长度被限制为 16 位。注意此限制仅影响 AC-Link 协议的 20 位数据时隙。对于接收时隙，输入数据将被简单地截取为 16 位。对于输出时隙，模块将数据字的最低 4 位设置为 0。时隙的这种截取，使得 ADC 和 DAC 数据被限制为 16 位，但允许 TXBUF 和 RXBUF 寄存器中恰当的数据对齐。每个 RXBUF 和 TXBUF 寄存器将存放一个数据时隙值。

© 2007 Microchip Technology Inc.

19.6.2 20 位 AC-LINK 模式

20 位 AC-Link 模式允许发送和接收数据时隙中的所有位，但不会保持 TXBUF 和 RXBUF 寄存器中的数据对齐。

除了帧同步信号的占空比不同之外，20 位 AC-Link 模式的工作方式类似于 DCI 模块的多通道模式。AC-Link 帧同步信号应该保持 16 个 CSCK 周期的高电平，并在接着的 240 个时钟周期内保持低电平。

20 位模式把每个 256 位 AC-Link 帧当作 16 个 16 位时隙处理。在 20 位 AC-Link 模式下，该模块像在 COFSG<3:0> = 1111 且 WS<3:0> = 1111 的条件下工作。20 位数据时隙的数据对齐无效。例如，通过将 TSCON 和 RSCON SFR 中的所有位置 1，整个 AC-Link 数据帧能够以打包的方式发送和接收。由于可用缓冲器的总长度为 64 位，所以传输该 AC-Link 帧需要经过 4 次连续的中断。应用软件必须跟踪当前 AC-Link 帧段。

19.7 I^2S 模式的操作

通过将值 01 写入 DCICON1 SFR 中的 COFSM<1:0> 控制位，将 DCI 模块配置为 I^2S 模式。在 I^2S 模式下工作时，DCI 模块将产生占空比为 50% 的帧同步信号。帧同步信号的每个边沿都将标记着一个新数据字传输的边界。

用户必须使用 DCICON2 SFR 中的 COFSG 和 WS 控制位选择帧长度和数据字长度。

19.7.1 I^2S 帧和数据字长度选择

设置 WS 和 COFSG 控制位，以便产生与 I^2S 数据帧长度的一半相等的周期。即帧长是左对齐或右对齐数据字传输所需的总的 CSCK 周期数。

必须设置 BLEN 位，从而得到所需的缓冲器长度。设置 BLEN<1:0> = 01 将产生 CPU 中断，每 I^2S 帧产生一次中断。

19.7.2 I^2S 数据对齐

根据 I^2S 规范，在默认情况下，将在 WS 信号跳变后过一个 CSCK 周期开始数据字的传输。可以使用 DCICON1 SFR 中的 DJST 控制位选择“最高有效位左对齐”选项。

如果 DJST = 1，则 I^2S 数据传输为最高有效位左对齐。数据字的最高有效位将在同一个 CSCK 周期内作为 COFS 信号的上升沿或下降沿出现在 CSDO 引脚上。在发送完数据字后， CSDO 引脚处于三态。

寄存器 19-1： DCICON1：DCI 控制寄存器 1

R/W-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
DCIEN	—	DCISIDL	—	DLOOP	CSCKD	CSCKE	COFSD
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
UNFM	CSDOM	DJST	—	—	—	COFSM<1:0>	
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **DCIEN**：DCI 模块使能位
1 = 使能模块
0 = 禁止模块

bit 14 **保留**：读为 0

bit 13 **DCISIDL**：DCI 在空闲模式停止控制位
1 = 模块在 CPU 空闲模式下将停止工作
0 = 模块在 CPU 空闲模式下将继续工作

bit 12 **保留**：读为 0

bit 11 **DLOOP**：数字环回模式控制位
1 = 使能数字环回模式， CSDI 和 CSDO 引脚在内部连接在一起。
0 = 禁止数字环回模式

bit 10 **CSCKD**：采样时钟方向控制位
1 = 当 DCI 模块使能时， CSCK 是输入引脚
0 = 当 DCI 模块使能时， CSCK 是输出引脚

bit 9 **CSCKE**：采样时钟边沿控制位
1 = 在串行时钟下降沿改变数据，在串行时钟上升沿采样数据
0 = 在串行时钟上升沿改变数据，在串行时钟下降沿采样数据

bit 8 **COFSD**：帧同步方向控制位
1 = 当 DCI 模块使能时， COFS 是输入引脚
0 = 当 DCI 模块使能时， COFS 是输出引脚

bit 7 **UNFM**：下溢模式位
1 = 在发送下溢时，发送最后写入发送寄存器的值
0 = 在发送下溢时，发送 0

bit 6 **CSDOM**：串行数据输出模式位
1 = 当发送时隙被禁止时， CSDO 引脚为三态
0 = 当发送时隙被禁止时， CSDO 引脚驱动为 0

bit 5 **DJST**：DCI 数据对齐控制位
1 = 在与帧同步脉冲相同的串行时钟周期内开始发送 / 接收数据
0 = 在帧同步脉冲的后一个串行时钟周期开始发送 / 接收数据

bit 4-2 **保留**：读为 0

bit 1-0 **COFSM<1:0>**：帧同步模式位
11 = 20 位 AC-Link 模式
10 = 16 位 AC-Link 模式
01 = I^2S 帧同步模式
00 = 多通道帧同步模式

© 2007 Microchip Technology Inc.

寄存器 19-2：　DCICON2：DCI 控制寄存器 2

U-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0	R/W-0
—	—	—	—	BLEN<1:0>		—	COFSG3
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
COFSG<2:0>			—	WS<3:0>			
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-12　**保留：**读为 0

bit 11-10　**BLEN<1:0>：**缓冲器长度控制位
11 = 在中断之间将缓冲四个数据字
10 = 在中断之间将缓冲三个数据字
01 = 在中断之间将缓冲两个数据字
00 = 在中断之间将缓冲一个数据字

bit 9　**保留：**读为 0

bit 8-5　**COFSG<3:0>：**帧同步发生器控制位
1111 = 数据帧有 16 个字
•
•
•
0010 = 数据帧有 3 个字
0001 = 数据帧有 2 个字
0000 = 数据帧有 1 个字

bit 4　**保留：**读为 0

bit 3-0　**WS<3:0>：**DCI 数据字长度位
1111 = 数据字长度为 16 位
•
•
•
0100 = 数据字长度为 5 位
0011 = 数据字长度为 4 位
0010 = **无效选择**。请勿使用。否则可能会产生无法预料的结果。
0001 = **无效选择**。请勿使用。否则可能会产生无法预料的结果。
0000 = **无效选择**。请勿使用。否则可能会产生无法预料的结果。

寄存器 19-3：　DCICON3：DCI 控制寄存器 3

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	BCG<11:8>			
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BCG<7:0>							
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-12　**保留：**读为 0

bit 11-0　**BCG<11:0>：**DCI 位时钟发生器控制位

© 2007 Microchip Technology Inc.

寄存器 19-4：　DCISTAT：DCI 状态寄存器

U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	SLOT<3:0>			
bit 15							bit 8

U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	ROV	RFUL	TUNF	TMPTY
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-12　**保留：**读为 0

bit 11-8　**SLOT<3:0>：**DCI 时隙状态位
1111 = Slot #15 目前有效
•
•
•
0010 = Slot #2 目前有效
0001 = Slot #1 目前有效
0000 = Slot #0 目前有效

bit 7-4　**保留：**读为 0

bit 3　**ROV：**接收溢出状态位
1 = 至少一个接收寄存器发生了接收溢出
0 = 没有发生接收溢出

bit 2　**RFUL：**接收缓冲器满状态位
1 = 在接收寄存器中有新数据
0 = 接收寄存器中为旧数据

bit 1　**TUNF：**发送缓冲器下溢状态位
1 = 至少一个发送寄存器发生了发送下溢
0 = 没有发生发送下溢

bit 0　**TMPTY：**发送缓冲器空状态位
1 = 发送寄存器为空
0 = 发送寄存器非空

寄存器 19-5：　　RSCON：DCI 接收时隙控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RSE15	RSE14	RSE13	RSE12	RSE11	RSE10	RSE9	RSE8
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RSE7	RSE6	RSE5	RSE4	RSE3	RSE2	RSE1	RSE0
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0　**RSE<15:0>**：接收时隙使能位
1 = 在相应的时隙 n 内接收 CSDI 数据
0 = 在相应的时隙 n 内忽略 CSDI 数据

寄存器 19-6：　　TSCON：DCI 发送时隙控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TSE15	TSE14	TSE13	TSE12	TSE11	TSE10	TSE9	TSE8
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TSE7	TSE6	TSE5	TSE4	TSE3	TSE2	TSE1	TSE0
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0　**TSE<15:0>**：发送时隙使能控制位
1 = 在相应的时隙 n 内发送发送缓冲器中的内容
0 = 根据 CSDOM 位的状态，在单个时隙内 CSDO 引脚呈现三态或驱动为逻辑 0

© 2007 Microchip Technology Inc.

20.0 10 位 /12 位模数转换器 （ADC）

注： 本数据手册总结了 dsPIC33FJXXXGPX06/X08/X10 器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

dsPIC33FJXXXGPX06/X08/X10 器件具有最多 32 路 ADC 输入通道。这些器件最多具有 2 个 ADC 模块（ADCx，其中 x = 1 或 2），每个模块都有一组特殊功能寄存器。

AD12B 位（ADxCON1<10>）允许用户对每个 ADC 模块进行配置，可以配置为 10 位，4 采样 / 保持 ADC（默认配置）或是 12 位， 1 采样 / 保持 ADC。

注： 在修改 AD12B 位前必须禁止 ADC 模块。

20.1 主要特性

10 位 ADC 配置具有以下主要特性：

- 逐次逼近 （Successive Approximation， SAR）转换
- 转换速度最高为 1.1 Msps
- 最多 32 个模拟输入引脚
- 外部参考电压输入引脚
- 可同时采样最多 4 个模拟输入引脚
- 自动通道扫描模式
- 可选的转换触发源
- 可选的缓冲器填充模式
- 4 种结果对齐选项（有符号 / 无符号，小数 / 整数）
- 可在 CPU 休眠和空闲模式下工作

12 位 ADC 配置支持所有上述特性，但以下情况除外：

- 在 12 位配置中，支持最大 500 ksps 的转换速度
- 在12位配置中只有1个采样/保持放大器，因此不支持多通道同时采样。

根据特定器件的引脚配置，ADC 最多有 32 个模拟输入引脚，指定为 AN0 到 AN31。此外，有两个可用于外部参考电压连接的模拟输入引脚。这两个参考电压输入可以与其他模拟输入引脚共用。实际的模拟输入引脚数和外部参考电压输入配置取决于具体的器件。请参见器件数据手册了解更多详细信息。

图 20-1 给出了 ADC 的框图。

20.2 ADC 初始化

应执行以下配置步骤。

1. 配置 ADC 模块：
   a) 选择端口引脚作为模拟输入引脚（ADxPCFGH<15:0> 或 ADxPCFGL<15:0>）
   b) 选择参考电压源以匹配模拟输入的预期范围（ADxCON2<15:13>）
   c) 选择模拟转换时钟以使期望的数据速率与处理器时钟匹配 （ADxCON3<5:0>）
   d) 确定使用多少路采样 / 保持通道（ADxCON2<9:8> 和 ADxPCFGH<15:0> 或 ADxPCFGL<15:0>）
   e) 选择适当的采样 / 转换序列（ADxCON1<7:5> 和 ADxCON3<12:8>）
   f) 选择转换结果在缓冲器中的存储方式（ADxCON1<9:8>）
   g) 开启 ADC 模块 （ADxCON1<15>）
2. 配置 ADC 中断 （如需要）：
   a) 清零 ADxIF 位
   b) 选择 ADC 中断优先级

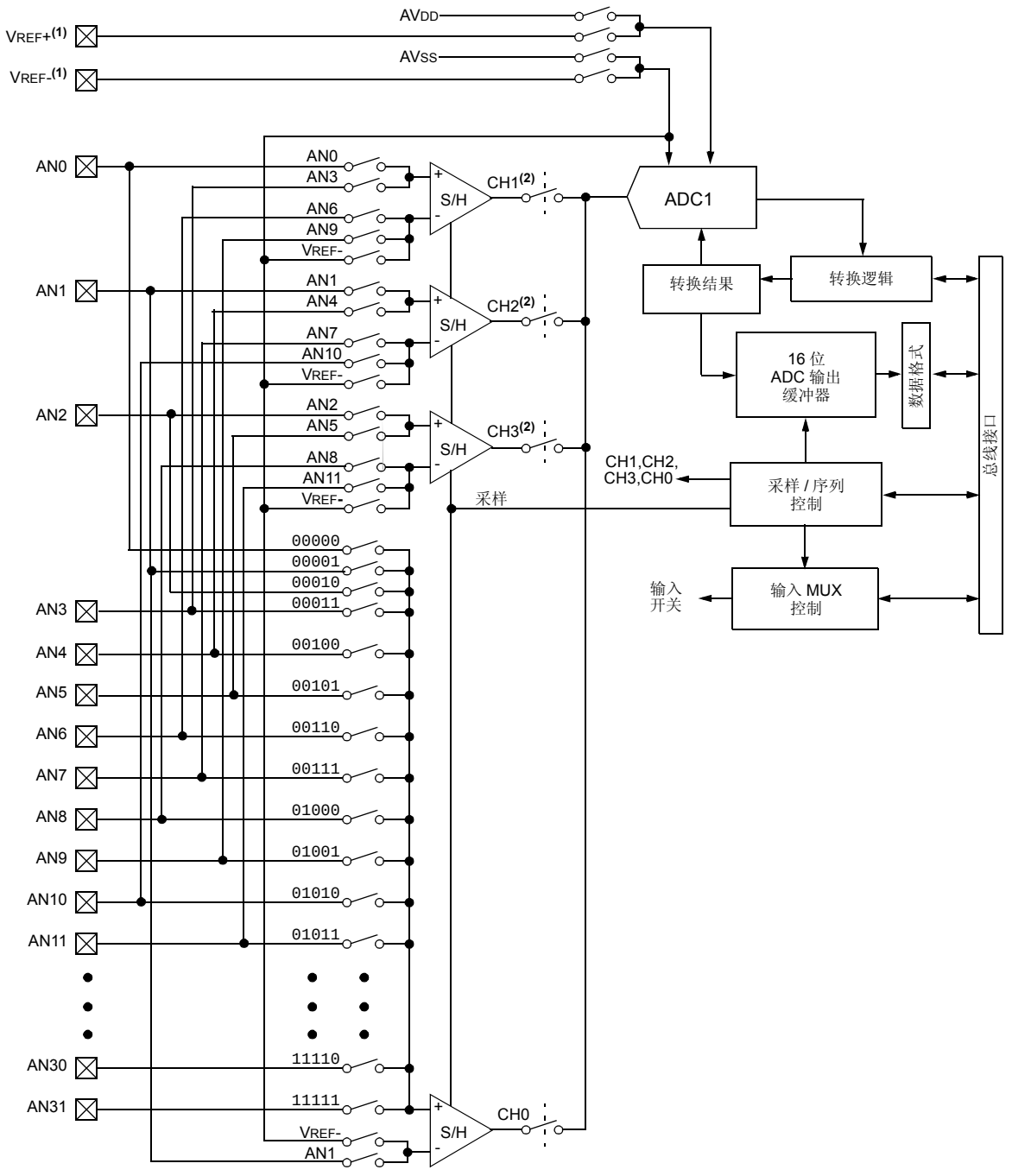
20.3 ADC 和 DMA

如果在触发一个中断之前，有多个转换结果需要被缓冲，就可使用 DMA 数据传输。 ADC1 和 ADC2 都能触发 DMA 数据传输。如果 ADC1 或 ADC2 被选作 DMA IRQ源，在由于ADC1或ADC2采样转换序列导致AD1IF或 AD2IF 位置 1 时，会发生 DMA 传输。

SMPI<3:0> 位（ADxCON2<5:2>）用来选择 DMA RAM 缓冲器指针增加的频率。

ADDMABM 位 （ADxCON1<12>）决定转换结果填充到 ADC 使用的 DMA RAM 缓冲器中的方式。如果该位置 1，则将数据以转换的顺序写入 DMA 缓冲器。模块将为DMA通道提供一个与非DMA独立缓冲器使用的地址相同的地址。如果 ADDMABM 位清零，那么 DMA 缓冲器以分散 / 集中模式写入数据。根据模拟输入的编号和 DMA 缓冲器的大小，模块为 DMA 通道提供分散 / 集中地址。

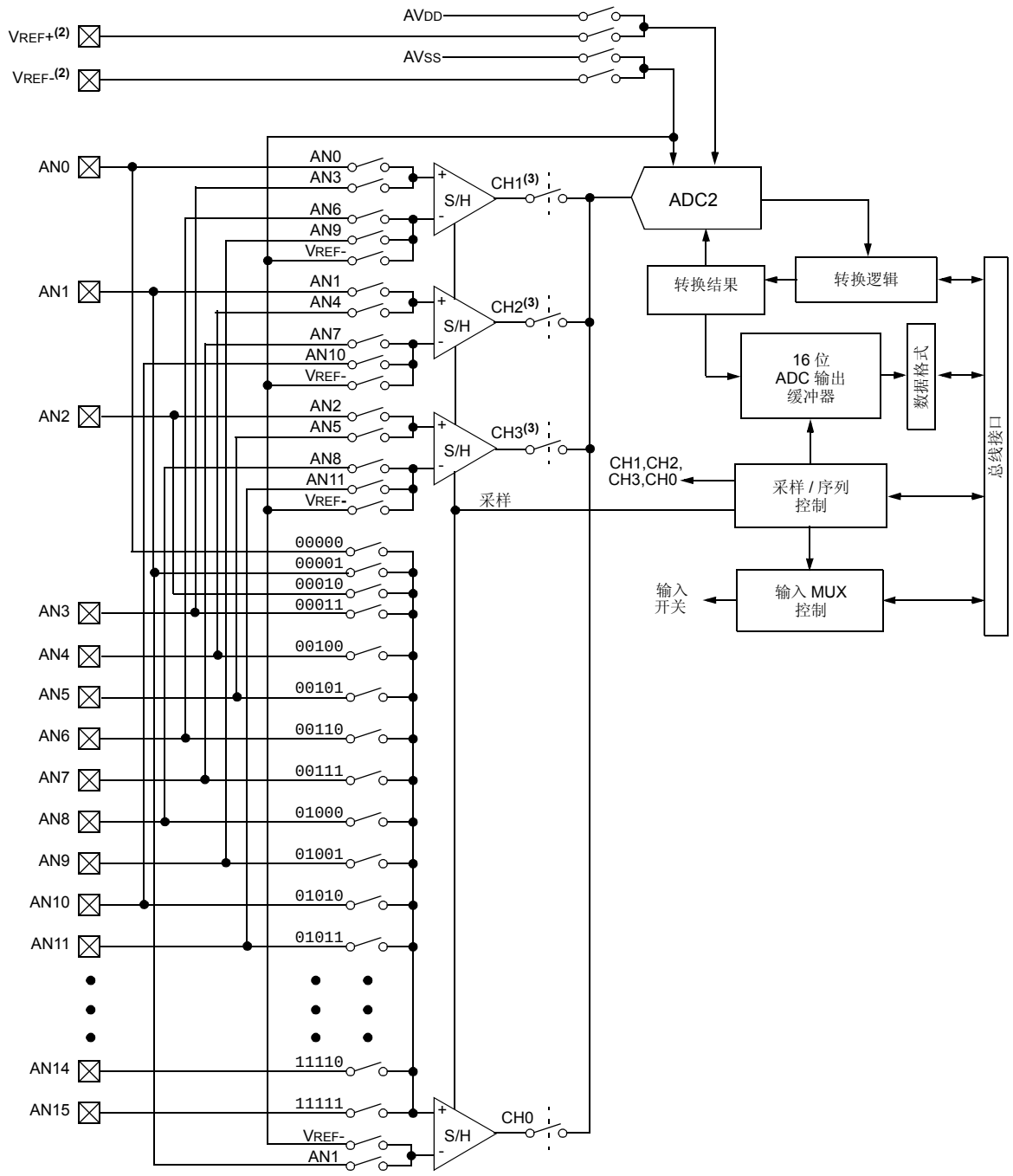
图 20-1：　　ADC1 模块框图



注　　1： VREF+ 和 VREF- 输入可以与其他模拟输入复用。请参见器件数据手册了解详细信息。
2： 通道 1、 2 和 3 不适用于 12 位工作模式。

© 2007 Microchip Technology Inc.

图 20-2： ADC2 模块框图 (1)



注 1： 对于具有 2 个 ADC 模块的器件， AN0-AN15 可以由 ADC1、 ADC2 或两者同时进行读取。

2： VREF+ 和 VREF- 输入可以与其他模拟输入复用。请参见器件数据手册了解详细信息。

3： 通道 1、 2 和 3 不适用于 12 位工作模式。

公式 20-1：　　ADC 转换时钟周期

$$T_{AD} = T_{CY}(ADCS + 1)$$

$$ADCS = \frac{T_{AD}}{T_{CY}} - 1$$

图 20-3：　　ADC 传递函数 （以 10 位 ADC 为例）

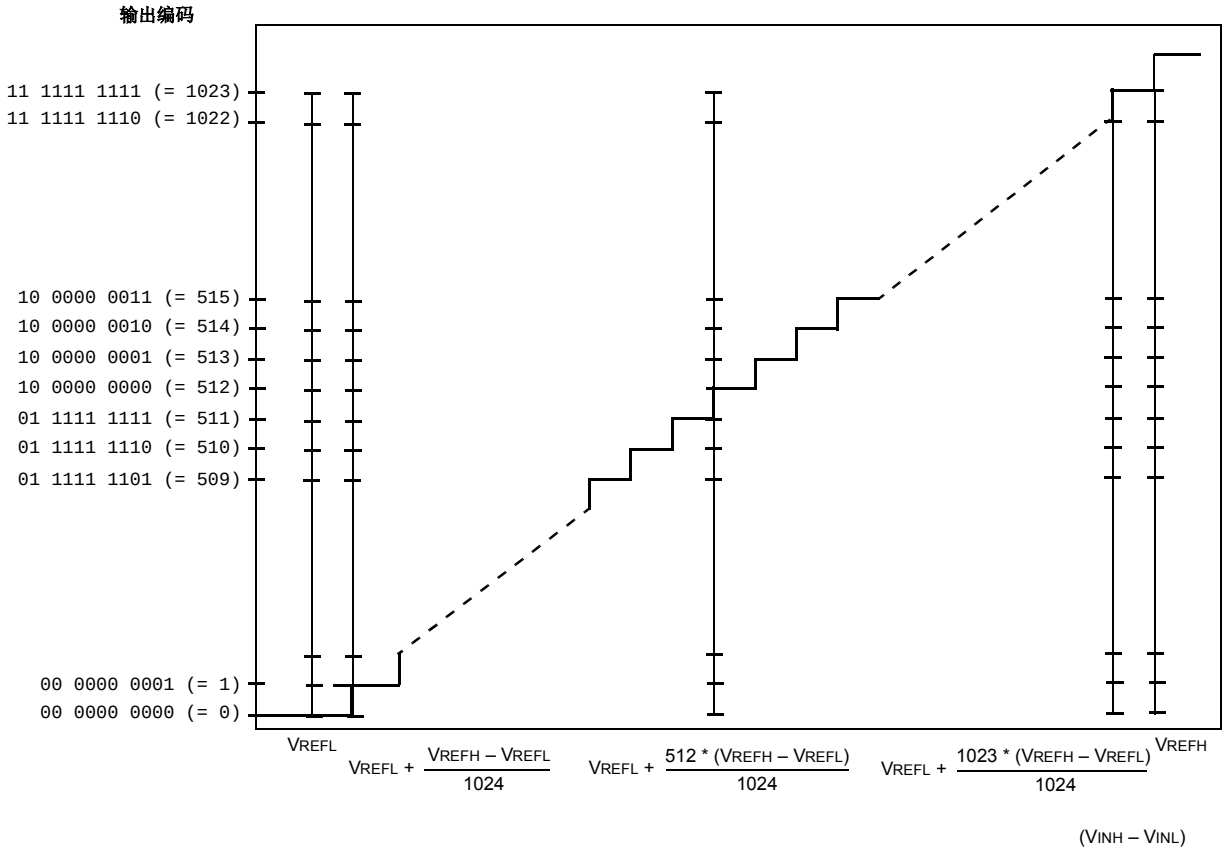
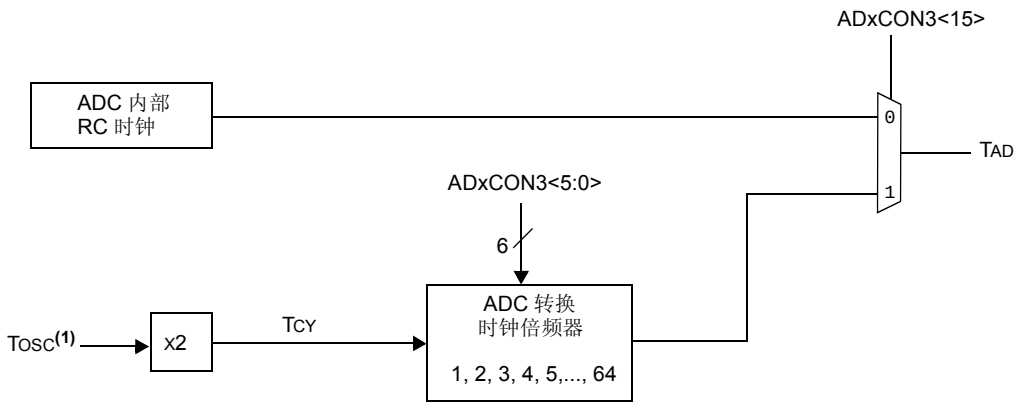


图 20-4：　　ADC 转换时钟周期框图



注：　　当使能 PLL 时，请参考图 8-2 来推导出 FOSC。如果没有使用 PLL，FOSC 与时钟源频率相等。TOSC = 1/FOSC。

© 2007 Microchip Technology Inc.

寄存器 20-1： ADxCON1：ADCx 控制寄存器 1（其中 x = 1 或 2）

R/W-0	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
ADON	—	ADSIDL	ADDMABM	—	AD12B	FORM<1:0>	
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0 HC,HS	R/C-0 HC, HS
SSRC<2:0>			—	SIMSAM	ASAM	SAMP	DONE
bit 7							bit 0

图注：	HC = 用硬件清零	HS = 用硬件置 1	
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **ADON**：ADC 工作模式位
1 = ADC 模块正在工作
0 = ADC 模块关闭

bit 14 **未实现**：读为 0

bit 13 **ADSIDL**：空闲模式停止位
1 = 当器件进入空闲模式时，模块停止工作
0 = 在空闲模式下模块继续工作

bit 12 **ADDMABM**：DMA 缓冲器构建模式位
1 = DMA 缓冲器以转换的顺序写入。模块将为 DMA 通道提供一个与非 DMA 独立缓冲器使用的地址相同的地址。
0 = DMA 缓冲器以分散 / 集中模式写入。根据模拟输入的编号和 DMA 缓冲器的大小，模块为 DMA 通道提供分散 / 集中地址。

bit 11 **未实现**：读为 0

bit 10 **AD12B**：10 位或 12 位工作模式位
1 = 12 位 1 通道 ADC 工作
0 = 10 位 4 通道 ADC 工作

bit 9-8 **FORM<1:0>**：数据输出格式位
对于 10 位工作：
11 = 有符号小数 （DOUT = sddd dddd dd00 0000，其中 s = d<9> 取反）
10 = 小数 （DOUT = dddd dddd dd00 0000）
01 = 有符号整数 （DOUT = ssss sssd dddd dddd，其中 s = d<9> 取反）
00 = 整数 （DOUT = 0000 00dd dddd dddd）
对于 12 位工作：
11 = 有符号小数 （DOUT = sddd dddd dddd 0000，其中 s = d<11> 取反）
10 = 小数 （DOUT = dddd dddd dddd 0000）
01 = 有符号整数 （DOUT = ssss sddd dddd dddd，其中 s = d<11> 取反）
00 = 整数 （DOUT = 0000 dddd dddd dddd）

bit 7-5 **SSRC<2:0>**：采样时钟源选择位
111 = 由内部计数器结束采样并启动转换 （自动转换）
110 = 保留
101 = 保留
100 = 保留
011 = 由 MPWM 间隔结束采样并启动转换
010 = 由 GP 定时器 （ADC1 采用 Timer3， ADC2 采用 Timer5）比较结束采样并启动转换
001 = 由 INTx 引脚的有效跳变结束采样并启动转换
000 = 由清零采样位结束采样并启动转换

bit 4 **未实现**：读为 0

寄存器 20-1： ADxCON1：ADCx 控制寄存器 1 （续）（其中 x = 1 或 2）

bit 3 **SIMSAM**：同步采样选择位 （仅当 CHPS<1:0> = 01 或 1x 时适用）
当 AD12B = 1 时， SIMSAM 为 U-0，未实现，读为 0
1 = 同时采样 CH0、 CH1、 CH2 和 CH3 （当 CHPS<1:0> = 1x 时）；或同时采样 CH0 和 CH1 （当 CHPS<1:0> = 01 时）
0 = 按顺序依次采样多路通道

bit 2 **ASAM**：ADC 采样自动启动位
1 = 最后一次转换结束后立即开始采样。 SAMP 位自动置 1。
0 = SAMP 位置 1 时开始采样

bit 1 **SAMP**：ADC 采样使能位
1 = ADC 采样 / 保持放大器正在采样
0 = ADC 采样 / 保持放大器保持采样结果
如果 ASAM = 0，由软件写入 1 开始采样。如果 ASAM = 1，该位由硬件自动置 1。
如果 SSRC = 000，由软件写入 0 结束采样并启动转换。如果 SSRC ≠ 000，由硬件自动清零来结束采样并启动转换。

bit 0 **DONE**：ADC 转换状态位
1 = ADC 转换完成
0 = ADC 转换尚未开始或在进行中
当 ADC 转换完成时，由硬件自动置 1。可由软件写入 0 来清零 DONE 状态位（不允许由软件写入 1）。清零该位不会影响进行中的任何操作。在新的转换开始时由硬件自动清零。

© 2007 Microchip Technology Inc.

寄存器 20-2： ADxCON2：ADCx 控制寄存器 2（其中 x = 1 或 2）

R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
VCFG<2:0>			—	—	CSCNA	CHPS<1:0>	
bit 15							bit 8

R-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BUFS	—	SMPI<3:0>				BUFM	ALTS
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-13 **VCFG<2:0>：**转换器参考电压配置位

	ADREF+	ADREF-
000	AVDD	AVSS
001	外部 VREF+	AVSS
010	AVDD	外部 VREF-
011	外部 VREF+	外部 VREF-
1xx	AVDD	Avss

bit 12-11 **未实现：**读为 0

bit 10 **CSCNA：**选择是否在使用采样多路开关 A 时扫描 CH0+ 输入的位
1 = 扫描输入
0 = 不扫描输入

bit 9-8 **CHPS<1:0>：**选择通道使用的位
当 AD12B = 1 时， CHPS<1:0> 为 U-0，未实现，读为 0
1x = 转换 CH0、 CH1、 CH2 和 CH3
01 = 转换 CH0 和 CH1
00 = 转换 CH0

bit 7 **BUFS：**缓冲器填充状态位 （仅当 BUFM = 1 时有效）
1 = ADC 当前在填充缓冲器的后半部分，用户应用程序应访问前半部分中的数据
0 = ADC 当前在填充缓冲器的前半部分，用户应用程序应访问后半部分中的数据

bit 6 **未实现：**读为 0

bit 5-2 **SMPI<3:0>：**选择 DMA 地址的递增速率或每次中断的采样 / 转换操作数的位
1111 = 每完成 16 个采样 / 转换操作将 DMA 地址递增 1 或产生中断
1110 = 每完成 15 个采样 / 转换操作将 DMA 地址递增 1 或产生中断
•
•
•
0001 = 每完成 2 个采样 / 转换操作将 DMA 地址递增 1 或产生中断
0000 = 每完成 1 个采样 / 转换操作将 DMA 地址递增 1 或产生中断

bit 1 **BUFM：**缓冲器填充模式选择位
1 = 在第一次中断发生时从缓冲器的前半部分开始填充，而在下一次中断发生时从后半部分开始填充
0 = 总是从前半部分开始填充缓冲器

bit 0 **ALTS：**备用输入采样模式选择位
1 = 在第一次采样时使用采样多路开关A选择的输入通道，而在下一次采样时使用采样多路开关B选择的输入通道
0 = 总是使用采样多路开关 A 选择的输入通道

寄存器 20-3： ADxCON3：ADCx 控制寄存器 3

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADRC	—	—	SAMC<4:0>				
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	ADCS<5:0>					
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **ADRC**：ADC 转换时钟源位
1 = ADC 内部 RC 时钟
0 = 时钟由系统时钟产生

bit 14-13 **未实现**：读为 0

bit 12-8 **SAMC<4:0>**：自动采样时间位
11111 = 31 TAD
•
•
•
00001 = 1 TAD
00000 = 0 TAD

bit 7-6 **未实现**：读为 0

bit 5-0 **ADCS<5:0>**：ADC 转换时钟选择位
111111 = TCY · (ADCS<7:0> + 1) = 64 · TCY = TAD
•
•
•
000010 = TCY · (ADCS<7:0> + 1) = 3 · TCY = TAD
000001 = TCY · (ADCS<7:0> + 1) = 2 · TCY = TAD
000000 = TCY · (ADCS<7:0> + 1) = 1 · TCY = TAD

© 2007 Microchip Technology Inc.

寄存器 20-4：　ADxCON4：ADCx 控制寄存器 4

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	DMABL<2:0>		
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-3　**未实现：**读为 0

bit 2-0　**DMABL<2:0>：**选择每路模拟输入的 DMA 缓冲单元数量的位

111 = 给每路模拟输入分配 128 字的缓冲区
110 = 给每路模拟输入分配 64 字的缓冲区
101 = 给每路模拟输入分配 32 字的缓冲区
100 = 给每路模拟输入分配 16 字的缓冲区
011 = 给每路模拟输入分配 8 字的缓冲区
010 = 给每路模拟输入分配 4 字的缓冲区
001 = 给每路模拟输入分配 2 字的缓冲区
000 = 给每路模拟输入分配 1 字的缓冲区

寄存器 20-5： ADxCHS123：ADCx 输入通道 1、 2 和 3 选择寄存器

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	CH123NB<1:0>		CH123SB
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	CH123NA<1:0>		CH123SA
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-11 **未实现：**读为 0

bit 10-9 **CH123NB<1:0>：**采样多路开关 B 的通道 1、 2 和 3 的反相输入选择位

当 AD12B = 1 时， CHxNB 为 U-0，未实现，读为 0

11 = CH1 的反相输入为 AN9， CH2 的反相输入为 AN10， CH3 的反相输入为 AN11

10 = CH1 的反相输入为 AN6， CH2 的反相输入为 AN7， CH3 的反相输入为 AN8

0x = CH1、 CH2 和 CH3 的反相输入都为 V_{REF-}

bit 8 **CH123SB：**采样多路开关 B 的通道 1、 2 和 3 的同相输入选择位

当 AD12B = 1 时， CHxSB 为 U-0，未实现，读为 0

1 = CH1 的同相输入为 AN3， CH2 的同相输入为 AN4， CH3 的同相输入为 AN5

0 = CH1 的同相输入为 AN0， CH2 的同相输入为 AN1， CH3 的同相输入为 AN2

bit 7-3 **未实现：**读为 0

bit 2-1 **CH123NA<1:0>：**采样多路开关 A 的通道 1、 2 和 3 的反相输入选择位

当 AD12B = 1 时， CHxNA 为 U-0，未实现，读为 0

11 = CH1 的反相输入为 AN9， CH2 的反相输入为 AN10， CH3 的反相输入为 AN11

10 = CH1 的反相输入为 AN6， CH2 的反相输入为 AN7， CH3 的反相输入为 AN8

0x = CH1、 CH2 和 CH3 的反相输入都为 V_{REF-}

bit 0 **CH123SA：**采样多路开关 A 的通道 1、 2 和 3 的同相输入选择位

当 AD12B = 1 时， CHxSA 为 U-0，未实现，读为 0

1 = CH1 的同相输入为 AN3， CH2 的同相输入为 AN4， CH3 的同相输入为 AN5

0 = CH1 的同相输入为 AN0， CH2 的同相输入为 AN1， CH3 的同相输入为 AN2

© 2007 Microchip Technology Inc.

寄存器 20-6： ADxCHS0：ADCx 输入通道 0 选择寄存器

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CH0NB	—	—	CH0SB<4:0>				
bit 15							bit 8

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CH0NA	—	—	CH0SA<4:0>				
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **CH0NB**：采样多路开关 B 的通道 0 的反相输入选择位
与 bit 7 定义相同。

bit 14-13 **未实现**：读为 0

bit 12-8 **CH0SB<4:0>**：采样多路开关 B 的通道 0 的同相输入选择位
与 bit<4:0> 定义相同。

bit 7 **CH0NA**：采样多路开关 A 的通道 0 的反相输入选择位
1 = 通道 0 的反相输入为 AN1
0 = 通道 0 的反相输入为 VREF-

bit 6-5 **未实现**：读为 0

bit 4-0 **CH0SA<4:0>**：采样多路开关 A 的通道 0 的同相输入选择位
11111 = 通道 0 的同相输入为 AN31
11110 = 通道 0 的同相输入为 AN30
•
•
•
00010 = 通道 0 的同相输入为 AN2
00001 = 通道 0 的同相输入为 AN1
00000 = 通道 0 的同相输入为 AN0

寄存器 20-7：　ADxCSSH：ADCx 输入扫描选择寄存器的高位字 (1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSS31	CSS30	CSS29	CSS28	CSS27	CSS26	CSS25	CSS24
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSS23	CSS22	CSS21	CSS20	CSS19	CSS18	CSS17	CSS16
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0　**CSS<31:16>**：ADC 输入扫描选择位
1 = 选择对 ANx 进行输入扫描
0 = 输入扫描时跳过 ANx

注　1： 对于没有 32 路模拟输入的器件，用户可以选择所有的 ADxCSSH 位。但是，如果器件上没有选择为进行扫描的相应输入，则将转换 ADREF-。

寄存器 20-8：　ADxCSSL：ADCx 输入扫描选择寄存器的低位字 (1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSS15	CSS14	CSS13	CSS12	CSS11	CSS10	CSS9	CSS8
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0　**CSS<15:0>**：ADC 输入扫描选择位
1 = 选择对 ANx 进行输入扫描
0 = 输入扫描时跳过 ANx

注　1： 对于没有 16 路模拟输入的器件，用户可以选择所有的 ADxCSSL 位。但是，如果器件上没有选择为进行扫描的相应输入，则将转换 ADREF-。

© 2007 Microchip Technology Inc.

寄存器 20-9： AD1PCFGH：ADC1 端口配置寄存器的高位字 (1,2)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG31	PCFG30	PCFG29	PCFG28	PCFG27	PCFG26	PCFG25	PCFG24
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG23	PCFG22	PCFG21	PCFG20	PCFG19	PCFG18	PCFG17	PCFG16
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0 **PCFG<31:16>**：ADC 端口配置控制位
1 = 端口引脚处于数字模式，使能端口读输入， ADC 输入多路开关连接到 AVSS
0 = 端口引脚处于模拟模式，禁止端口读输入， ADC 采样引脚电压

注 1： 对于没有 32 路模拟输入的器件，所有 PCFG 位都由用户读 / 写。但是，如果器件上不含相应的输入，则 PCFG 位被忽略。
2： ADC2 仅支持模拟输入 AN0-AN15 ；因此，不存在 ADC2 端口配置寄存器。

寄存器 20-10： ADxPCFGL：ADCx 端口配置寄存器的低位字 (1,2)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0
bit 7							bit 0

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0 **PCFG<15:0>**：ADC 端口配置控制位
1 = 端口引脚处于数字模式，使能端口读输入， ADC 输入多路开关连接到 AVSS
0 = 端口引脚处于模拟模式，禁止端口读输入， ADC 采样引脚电压

注 1： 对于没有 16 路模拟输入的器件，所有 PCFG 位都由用户读 / 写。但是，如果器件上不含相应的输入，则 PCFG 位被忽略。
2： 对于具有 2 个模数转换模块的器件， AD1PCFGL 和 AD2PCFGL 都会影响与 AN0-AN15 复用的端口引脚的配置。

注：

© 2007 Microchip Technology Inc.

21.0 特殊功能

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

dsPIC33FJXXXGPX06/X08/X10 器件具有几项特殊的功能旨在最大限度地提高应用的灵活性和可靠性，并通过减少外部组件的使用将成本降至最低。提供的特殊功能包括：

- 灵活的配置
- 看门狗定时器 （WDT）
- 代码保护和 CodeGuard™ 安全性
- JTAG 边界扫描接口
- 在线串行编程 （ICSP™）
- 在线仿真

21.1 配置位

可以通过对配置位编程 （读为 0）或不编程 （读为 1）来选择不同的器件配置。这些配置位被映射到程序存储器以 0xF80000 开始的单元中。

器件配置寄存器的映射如表 21-1 所示。

表 21-2 给出了 FBS、FSS、FGS、FOSCSEL、FOSC、FWDT、FPOR 和 FICD 配置寄存器中各个配置位的说明。

注意，地址 0xF80000 超出了用户程序存储空间。事实上，它属于只能使用表读和表写访问的配置存储空间（0x800000-0xFFFFFF）。

所有器件配置寄存器的高字节应该总为 1111 1111。这使得当极少事件意外执行这些存储单元时将其作为 NOP 指令来执行。由于没有在相应的存储单元中实现这些配置位，因此向这些存储单元写入 1 不会影响器件工作。

为了避免在代码执行期间配置被意外改变，所有的可编程配置位只可被写入一次。在上电周期内对位进行初始化编程之后就不能再次写入该位了。改变器件的配置需要对器件重复上电。

表 21-1： 器件配置寄存器的映射

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xF80000	FBS	RBS<1:0>		—	—	BSS<2:0>			BWRP
0xF80002	FSS	RSS<1:0>		—	—	SSS<2:0>			SWRP
0xF80004	FGS	—	—	—	—	—	GSS1	GSS0	GWRP
0xF80006	FOSCSEL	IESO	—	—	—	—	FNOSC<2:0>		
0xF80008	FOSC	FCKSM<1:0>		—	—	—	OSCIOFNC	POSCMD<1:0>	
0xF8000A	FWDT	FWDTEN	WINDIS	—	WDTPRE	WDTPOST<3:0>			
0xF8000C	FPOR	—	—	—	—	—	FPWRT<2:0>		
0xF8000E	RESERVED3	保留 (1)							
0xF80010	FUID0	用户部件 ID 字节 0							
0xF80012	FUID1	用户部件 ID 字节 1							
0xF80014	FUID2	用户部件 ID 字节 2							
0xF80016	FUID3	用户部件 ID 字节 3							

注 1： 这些保留的位读为 1 并且必须被编程为 1。

2： 未实现位，读为 0。

表 21-2： dsPIC33FJXXXGPX06/X08/X10 配置位的说明

位域	寄存器	说明
BWRP	FBS	引导段程序闪存写保护 1 = 引导段可写 0 = 引导段被写保护
BSS<2:0>	FBS	引导段程序闪存代码保护大小 X11 = 无引导程序闪存段 引导空间为 1K 指令字减去 VS 大小 110 = 标准安全性；引导程序闪存段开始于 VS 末端，结束于 0007FEh 010 = 高安全性；引导程序闪存段开始于 VS 末端，结束于 0007FEh 引导空间为 4K 指令字减去 VS 大小 101 = 标准安全性；引导程序闪存段开始于 VS 末端，结束于 001FFEh 001 = 高安全性；引导程序闪存段开始于 VS 末端，结束于 001FFEh 引导空间为 8K 指令字减去 VS 大小 100 = 标准安全性；引导程序闪存段开始于 VS 末端，结束于 003FFEh 000 = 高安全性；引导程序闪存段开始于 VS 末端，结束于 003FFEh
RBS<1:0>	FBS	引导段 RAM 代码保护 10 = 未定义引导 RAM 10 = 引导 RAM 为 128 字节 01 = 引导 RAM 为 256 字节 00 = 引导 RAM 为 1024 字节
SWRP	FSS	安全段程序闪存写保护 1 = 安全段可写 0 = 安全段被写保护

© 2007 Microchip Technology Inc.

表 21-2： dsPIC33FJXXXGPX06/X08/X10 配置位的说明 （续）

位域	寄存器	说明
SSS<2:0>	FSS	安全段程序闪存代码保护大小 **（对于 128K 和 256K 器件）** X11 = 无安全程序闪存段 安全空间为 8K 指令字减去 BS 大小 110 = 标准安全性；安全程序闪存段开始于 BS 末端，结束于 0x003FFE 010 = 高安全性；安全程序闪存段开始于 BS 末端，结束于 0x003FFE 安全空间为 16K 指令字减去 BS 大小 101 = 标准安全性；安全程序闪存段开始于 BS 末端，结束于 0x007FFE 001 = 高安全性；安全程序闪存段开始于 BS 末端，结束于 0x007FFE 安全空间为 32K 指令字减去 BS 大小 100 = 标准安全性；安全程序闪存段开始于 BS 末端，结束于 0x00FFFE 000 = 高安全性；安全程序闪存段开始于 BS 末端，结束于 0x00FFFE **（对于 64K 器件）** X11 = 无安全程序闪存段 安全空间为 4K 指令字减去 BS 大小 110 = 标准安全性；安全程序闪存段开始于 BS 末端，结束于 0x001FFE 010 = 高安全性；安全程序闪存段开始于 BS 末端，结束于 0x001FFE 安全空间为 8K 指令字减去 BS 大小 101 = 标准安全性；安全程序闪存段开始于 BS 末端，结束于 0x003FFE 001 = 高安全性；安全程序闪存段开始于 BS 末端，结束于 0x003FFE 安全空间为 16K 指令字减去 BS 大小 100 = 标准安全性；安全程序闪存段开始于 BS 末端，结束于 007FFEh 000 = 高安全性；安全程序闪存段开始于 BS 末端，结束于 0x007FFE
RSS<1:0>	FSS	安全段 RAM 代码保护 10 = 未定义安全 RAM 10 = 安全 RAM 为 256 字节减去 BS RAM 大小 01 = 安全 RAM 为 2048 字节减去 BS RAM 大小 00 = 安全 RAM 为 4096 字节减去 BS RAM 大小
GSS<1:0>	FGS	通用段代码保护位 11 = 用户程序存储器不被代码保护 10 = 标准安全性；通用程序闪存段开始于 SS 末端，结束于 EOM 0x = 高安全性；通用程序闪存段开始于 SS 末端，结束于 EOM
GWRP	FGS	通用段写保护位 1 = 用户程序存储器不被写保护 0 = 用户程序存储器被写保护

表 21-2： dsPIC33FJXXXGPX06/X08/X10 配置位的说明 （续）

位域	寄存器	说明
IESO	FOSCSEL	双速振荡器启动使能位 1 = 使用 FRC 启动器件，然后自动切换到就绪的用户选择的振荡器源 0 = 使用用户选择的振荡器源启动器件
FNOSC<2:0>	FOSCSEL	初始振荡器源选择位 111 = 带后分频器的内部快速 RC （FRC）振荡器 110 = 带 16 分频的内部快速 RC （FRC）振荡器 101 = LPRC 振荡器 100 = 辅助 （LP）振荡器 011 = 带 PLL 的主 （XT、 HS 或 EC）振荡器 010 = 主 （XT、 HS 或 EC）振荡器 001 = 带 PLL 的内部快速 RC （FRC）振荡器 000 = FRC 振荡器
FCKSM<1:0>	FOSC	时钟切换模式位 1x = 禁止时钟切换，禁止故障保护时钟监视器 01 = 使能时钟切换，禁止故障保护时钟监视器 00 = 使能时钟切换，使能故障保护时钟监视器
OSCIOFNC	FOSC	OSC2 引脚功能位 （XT 和 HS 模式除外） 1 = OSC2 为时钟输出 0 = OSC2 为通用数字 I/O 引脚
POSCMD<1:0>	FOSC	主振荡器模式选择位 11 = 禁止主振荡器 10 = HS 晶振模式 01 = XT 晶振模式 00 = EC （外部时钟）模式
FWDTEN	FWDT	看门狗定时器使能位 1 = 始终使能看门狗定时器 （不能禁止 LPRC 振荡器。将 RCON 寄存器中的 SWDTEN 位清零不会产生任何影响。） 0 = 通过用户软件使能 / 禁止看门狗定时器 （可通过清零 RCON 寄存器中的 SWDTEN 位来禁止 LPRC）
WINDIS	FWDT	看门狗定时器窗口使能位 1 = 非窗口模式下的看门狗定时器 0 = 窗口模式下的看门狗定时器
WDTPRE	FWDT	看门狗定时器预分频比位 1 = 1:128 0 = 1:32
WDTPOST	FWDT	看门狗定时器后分频比位 1111 = 1:32,768 1110 = 1:16,384 . . . 0001 = 1:2 0000 = 1:1
保留	RESERVED3, FPOR	保留 （读为 1 和写为 1，或读为 0 和写为 0）
—	FGS, FOSCSEL, FOSC, FWDT, FPOR	未实现 （读为 0，写为 0）

© 2007 Microchip Technology Inc.

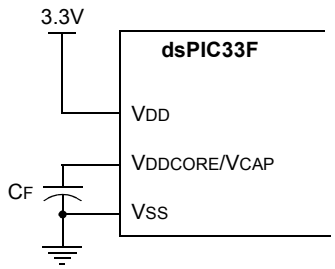
21.2 片上稳压器

所有的 dsPIC33FJXXXGPX06/X08/X10 器件使用标称值为 2.5V 的电压为其内核数字逻辑供电。对于需要工作在一个更高的典型电压值如 3.3V 的设计中，这可能会产生冲突。为简化系统设计，dsPIC33FJXXXGPX06/X08/X10 系列中的所有器件均包含一个片上稳压器，可使器件内核逻辑在 VDD 下工作。

稳压器通过其他 VDD 引脚向内核供电。当使能了稳压器时，必须将一个低 ESR（小于 5 欧姆）电容（如钽电容或陶瓷电容）连接到 VDDCORE/VCAP 引脚（图 21-1）。这可帮助维持稳压器的稳定性。滤波电容的推荐值在**第 24.1 节 “直流特性”**的**表 24-13：“内部稳压器规范”**中提供。

在 POR 时，片上稳压器需要约 20 μs 的时间产生输出电压。这段时间被称为 TSTARTUP，在这期间禁止代码执行。在每次掉电后器件恢复工作的过程中都将有一段 TSTARTUP 时间。

图 21-1： 片上稳压器的连接 (1)



注 1： 这些为典型的工作电压。关于 VDD 和 VDDCORE 的全部工作电压范围，请参见**第 24.1 节 “直流特性”**的**表 24-13：“内部稳压器规范”**。

21.3 BOR：欠压复位

BOR（欠压复位）模块是基于内部参考电压电路的，该电路监视经过稳压的电压 VDDCORE。BOR 模块的主要用途是在发生欠压条件时产生器件复位。欠压条件通常由交流电源线上的干扰信号（即，由于电源传输线路不良造成的交流周期波形丢失，或者由于接入大感性负载时电流消耗过大造成电压下降）产生。

BOR 将产生复位器件的复位脉冲。BOR 会根据器件配置位（FNOSC<2:0> 和 POSCMD<1:0>）的值选择时钟源。此外，如果选择了振荡器模式，BOR 将激活振荡器起振定时器（OST）。系统时钟将保持到 OST 超时。如果使用了 PLL，则时钟将被保持到 LOCK 位（OSCCON<5>）置 1。

同时，将在内部复位释放前应用 PWRT 延时（TPWRT）。如果 TPWRT = 0 且使用了晶振，那么会应用 TFSCM = 100 的标称延时。这种情况下总的延时为 TFSCM。

BOR状态位（RCON<1>）将置1，以表明发生了BOR。BOR 电路使能时，将在休眠或空闲模式下继续工作，当 Vdd 下降到 BOR 门限电压以下时将复位器件。

21.4 看门狗定时器 （WDT）

对于 dsPIC33FJXXXGPX06/X08/X10 器件， WDT 由 LPRC 振荡器驱动。当使能 WDT 时，时钟源也将使能。

由 LPRC 提供的 WDT 时钟源的频率标称值为 32 kHz。这一频率的信号可以输入给可配置为 5 位 （32 分频）或 7 位 （128 分频）操作的预分频器。预分频比由 WDTPRE 配置位设置。使用 32 kHz 输入，预分频器在 5 位模式下将产生一个 1 ms 的标称 WDT 超时周期（TWDT），或在 7 位模式下超时周期为 4 ms。

分频比可变的后分频器对 WDT 预分频器的输出进行分频，并扩展超时周期范围。后分频比由 WDTPOST<3:0> 配置位（FWDT<3:0>）控制，该配置位共允许选择 16 种设置，从 1:1 到 1:32,768。使用预分频器和后分频器，可以使超时周期的范围扩展到1 ms 至 131 秒。

WDT、预分频器和后分频器在以下条件下复位：

- 任何器件复位时
- 在完成时钟切换后，无论时钟切换是由软件 （即，在更改 NOSC 位后将 OSWEN 位置 1）引起还是由硬件 （即，故障保护时钟监视器）引起的
- 当执行 PWRSAV 指令时 （即，进入休眠或空闲模式）
- 当器件退出休眠或空闲模式恢复正常工作时
- 当在正常执行过程中使用 CLRWDT 指令时

如果使能 WDT，它将在休眠或空闲模式下继续运行。当发生 WDT 超时时，将唤醒器件并且代码将继续从 PWRSAV 指令处开始执行。当器件唤醒后，需要用软件将相应的 SLEEP 或 IDLE 位 （RCON<3,2>）清零。

WDT 标志位 WDTO （RCON<4>）不会在 WDT 超时后自动清零。要检测后面的 WDT 事件，必须用软件将该标志位清零。

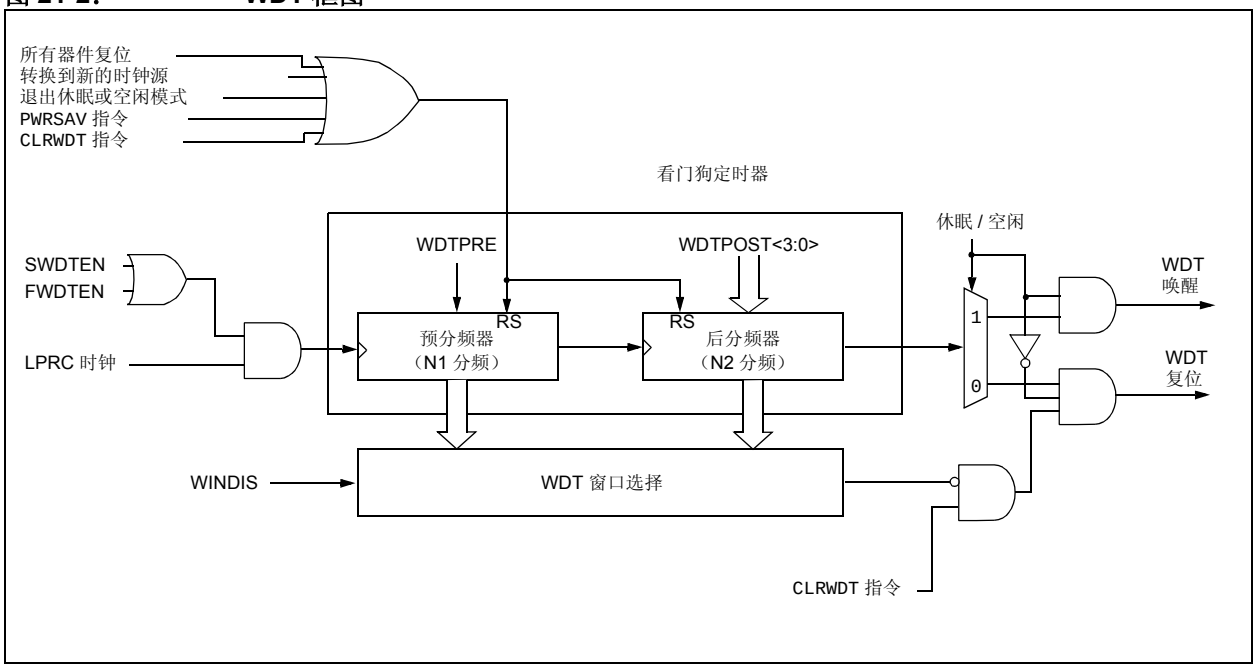
注：	执行 CLRWDT 和 PWRSAV 指令会将预分频器和后分频器的计数值清零。

WDT 的使能或禁止由 FWDT 配置寄存器中的 FWDTEN 配置位控制。当 FWDTEN 配置位置 1 时，WDT 始终是使能的。

当 FWDTEN 配置位被编程为 0 时，可以选择用软件控制 WDT。通过将 SWDTEN 控制位 （RCON<5>）置 1 用软件使能 WDT。SWDTEN 控制位在任何器件复位时清零。软件 WDT 选项允许用户在关键代码段使能 WDT 并在非关键代码段禁止 WDT，从而最大限度地降低功耗。

注：	如果 WINDIS 位 （FWDT<6>）清零，CLRWDT 指令应仅在 WDT 周期的最后 1/4 中被应用软件执行。该 CLRWDT 窗口可通过使用定时器确定。如果在该窗口之前执行 CLRWDT 指令，将会使 WDT 复位。

图 21-2： WDT 框图



© 2007 Microchip Technology Inc.

21.5 JTAG 接口

dsPIC33FJXXXGPX06/X08/X10 器件实现了一个 JTAG 接口，该接口支持边界扫描器件测试以及在线编程。关于该接口的详细信息将会在以后的文档版本中提供。

21.6 代码保护和 CodeGuard™ 安全性

dsPIC33F 系列产品提供了 CodeGuard™ 安全性的高级实现。CodeGuard 安全性允许多方安全地共用单个芯片上的资源（存储器、中断和外设）。这一功能有助于在协同系统设计中保护各方的知识产权（Intellectual Property，IP）。

CodeGuard™ 安全性与软件加密函数库配合使用时，即使在单个芯片上存在多个知识产权（IP），也可以使用 CodeGuard 安全性来安全地更新闪存。代码保护功能随所实现的实际 dsPIC33F 的不同而有所不同。后续章节将对这些功能进行概述。

代码保护功能由配置寄存器 FBS、FSS 和 FGS 控制。

注：	请参见《CodeGuard™ 安全》参考章节（DS70180A_CN）获得有关 CodeGuard 安全性使用、配置和操作方面的更多信息。

21.7 在线串行编程

可以在最终的应用电路中对 dsPIC33FJXXXGPX06/X08/X10 系列数字信号控制器进行串行编程。只需要 5 根线即可实现这一操作，其中时钟线、数据线各一根，其余 3 根分别是电源线、接地线和编程电压线。串行编程允许用户在生产电路板时使用未编程器件，仅在产品交付之前才对数字信号控制器进行编程，从而可以使用最新版本的固件或者定制固件进行编程。请参见《dsPIC33F/PIC24H 闪存编程规范》（DS70152C_CN）文档了解有关 ICSP 的详细信息。

可使用 3 对编程时钟 / 数据引脚中的任意一对：

- PGC1/EMUC1 和 PGD1/EMUD1
- PGC2/EMUC2 和 PGD2/EMUD2
- PGC3/EMUC3 和 PGD3/EMUD3

21.8 在线调试器

当选择 MPLAB® ICD 2 作为调试器时，使能在线调试功能。该功能允许与 MPLAB IDE 配合使用进行简单的调试。通过 EMUCx（仿真 / 调试时钟）和 EMUDx（仿真 / 调试数据）引脚功能控制调试功能。

可使用 3 对调试时钟 / 数据引脚中的任意一对：

- PGC1/EMUC1 和 PGD1/EMUD1
- PGC2/EMUC2 和 PGD2/EMUD2
- PGC3/EMUC3 和 PGD3/EMUD3

要使用器件的在线调试功能，就必须在设计中正确对 MCLR、V_{DD}、V_{SS}、PGC、PGD 和 EMUDx/EMUCx 引脚对进行 ICSP 连接。此外，当使能该功能时，某些资源就不能用于一般用途了。这些资源包括数据 RAM 的前 80 字节和两个 I/O 引脚。

注：

© 2007 Microchip Technology Inc.

22.0 指令集汇总

注： 本数据手册总结了dsPIC33FJXXXGPX06/X08/X10 器件的功能。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F 系列参考手册》。请参见 Microchip 网站（www.microchip.com）了解最新的《dsPIC33F 系列参考手册》章节。

dsPIC33F 系列器件的指令集与 dsPIC30F 系列器件的指令集相同。

大部分指令的长度为一个程序存储字（24 位）。只有三条指令需要两个程序存储单元。

每一条单字指令长 24 位，分为一个指定指令类型的 8 位操作码和进一步指定指令操作的一个或多个操作数。

指令集是高度正交的，分为 5 个基本类别：

- 针对字或字节的操作
- 针对位的操作
- 立即数操作
- DSP 操作
- 控制操作

表 22-1 给出了在说明指令时使用的通用符号。

表 22-2 是 dsPIC33F 指令集汇总，还给出了每条指令影响的状态标志位。

大多数针对字或字节的 W 寄存器指令（包括桶形移位指令）有三个操作数：

- 第一个源操作数通常是寄存器 Wb，不带任何地址修改量
- 第二个源操作数通常是寄存器 Ws，带或不带地址修改量
- 运算结果的目标寄存器，通常是寄存器 Wd，带或不带地址修改量

不过，针对字或字节的文件寄存器指令只有两个操作数：

- 文件寄存器由 f 值指定
- 目标寄存器可以是文件寄存器 f 或 W0 寄存器（用 WREG 表示）

大多数位操作类指令（包括简单的循环 / 移位指令）有两个操作数：

- W 寄存器（带或不带地址修改量）或文件寄存器（由 Ws 或 f 的值指定）
- W 寄存器或文件寄存器中的位（由一个立即数指定，或者由寄存器 Wb 的内容间接指定）

涉及数据传送的立即数指令，可以使用下列操作数：

- 要被装入到 W 寄存器或文件寄存器中的立即数（由 k 值指定）
- 要装入立即数的 W 寄存器或文件寄存器（由 Wb 或 f 指定）

然而，涉及算术或逻辑运算的立即数指令，使用如下的操作数：

- 第一个源操作数是寄存器 Wb，不带任何地址修改量
- 第二个源操作数是立即数
- 操作结果的目标寄存器（仅在与第一个源操作数不同时）通常是寄存器 Wd（带或不带地址修改量）

MAC 类 DSP 指令可使用下列操作数：

- 要使用的累加器（A 或 B）（必需的操作数）
- 要用作两个操作数的 W 寄存器
- X 和 Y 地址空间预取操作
- X 和 Y 地址空间预取目标寄存器
- 累加器回写目标寄存器

与乘法无关的其他 DSP 指令使用的操作数可能包括：

- 要使用的累加器（必需）
- 源操作数或目标操作数（分别由 Wso 或 Wdo 指定），带或不带地址修改量
- 移位位数，由 W 寄存器 Wn 或立即数指定

控制指令可以使用下列操作数：

- 程序存储地址
- 表读和表写指令的模式

除了某些双字指令外，所有指令都是单字指令；双字指令之所以是双字长的 （48 位），是因为要用 48 位来提供所需信息。在第二个字中，8 个 MSb 全为 0。如果指令自身把第二个字当作一条指令来执行的话，它将作为一条 NOP 指令来执行。

大多数单字长指令都在一个指令周期内执行，除非条件测试为真或者指令执行结果改变了程序计数器。对于上述两种特殊情况，指令执行需要两个指令周期，在第二个指令周期中执行一条 NOP 指令。值得注意的例外是 BRA（无条件 / 计算转移）、间接 CALL/GOTO、所有的表读和表写以及 RETURN/RETFIE 指令，它们是单字长指令，但执行需要两个或三个周期。某些与跳过后续指令有关的指令，如果要执行跳过的话，可能需要两个或三个周期，这取决于被跳过的指令是单字还是双字指令。此外，双字传送需要两个周期。双字指令执行需要两个指令周期。

注： 有关指令集的更多详细信息，请参见《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

表 22-1： 操作码说明中使用的符号

字段	说明
#text	表示由 text 定义的立即数
(text)	表示 text 的内容
[text]	表示由 text 寻址的存储单元
{ }	可选字段或操作
<n:m>	寄存器位域
.b	字节模式选择
.d	双字模式选择
.S	影子寄存器选择
.w	字模式选择 （默认）
Acc	两个累加器 {A, B} 之一
AWB	累加器回写目标地址寄存器 ∈ {W13, [W13]+=2}
bit4	4 位位选择字段 （用于字寻址指令） ∈ {0...15}
C, DC, N, OV, Z	MCU 状态位：进位、半进位、负、溢出和全零标志位
Expr	绝对地址、标号或表达式 （由链接器解析）
f	文件寄存器地址 ∈ {0x0000...0x1FFF}
lit1	1 位无符号立即数 ∈ {0,1}
lit4	4 位无符号立即数 ∈ {0...15}
lit5	5 位无符号立即数 ∈ {0...31}
lit8	8 位无符号立即数 ∈ {0...255}
lit10	10 位无符号立即数，对于字节模式， ∈ {0...255}；对于字模式， ∈ {0:1023}
lit14	14 位无符号立即数 ∈ {0...16384}
lit16	16 位无符号立即数 ∈ {0...65535}
lit23	23 位无符号立即数 ∈ {0...8388608}； LSb 必须为 0
None	字段无需内容，可为空
OA, OB, SA, SB	DSP 状态位：AccA 溢出、 AccB 溢出、 AccA 饱和和 AccB 饱和
PC	程序计数器
Slit10	10 位有符号立即数 ∈ {-512...511}
Slit16	16 位有符号立即数 ∈ {-32768...32767}
Slit6	6 位有符号立即数 ∈ {-16...16}
Wb	基本 W 寄存器 ∈ {W0..W15}
Wd	目标 W 寄存器 ∈ {Wd, [Wd], [Wd++], [Wd--], [++Wd], [--Wd]}
Wdo	目标 W 寄存器 ∈ {Wnd, [Wnd], [Wnd++], [Wnd--], [++Wnd], [--Wnd], [Wnd+Wb]}
Wm,Wn	被除数和除数工作寄存器对 （直接寻址）

© 2007 Microchip Technology Inc.

表 22-1： 操作码说明中使用的符号 （续）

字段	说明
Wm*Wm	用于平方指令的被乘数和乘数工作寄存器对 ∈ {W4 * W4,W5 * W5,W6 * W6,W7 * W7}
Wm*Wn	用于 DSP 指令的被乘数和乘数工作寄存器对 ∈ {W4 * W5,W4 * W6,W4 * W7,W5 * W6,W5 * W7,W6 * W7}
Wn	16 个工作寄存器中之一 ∈ {W0..W15}
Wnd	16 个目标工作寄存器之一 ∈ {W0..W15}
Wns	16 个源工作寄存器之一 ∈ {W0..W15}
WREG	W0 （文件寄存器指令中使用的工作寄存器）
Ws	源 W 寄存器 ∈ {Ws, [Ws], [Ws++], [Ws--], [++Ws], [--Ws]}
Wso	源 W 寄存器 ∈ {Wns, [Wns], [Wns++], [Wns--], [++Wns], [--Wns], [Wns+Wb]}
Wx	用于 DSP 指令的 X 数据空间预取地址寄存器 ∈ {[W8]+=6, [W8]+=4, [W8]+=2, [W8], [W8]-=6, [W8]-=4, [W8]-=2, [W9]+=6, [W9]+=4, [W9]+=2, [W9], [W9]-=6, [W9]-=4, [W9]-=2, [W9 + W12], 无 }
Wxd	用于 DSP 指令的 X 数据空间预取目标寄存器 ∈ {W4..W7}
Wy	用于 DSP 指令的 Y 数据空间预取地址寄存器 ∈ {[W10]+=6, [W10]+=4, [W10]+=2, [W10], [W10]-=6, [W10]-=4, [W10]-=2, [W11]+=6, [W11]+=4, [W11]+=2, [W11], [W11]-=6, [W11]-=4, [W11]-=2, [W11 + W12], 无 }
Wyd	用于 DSP 指令的 Y 数据空间预取目标寄存器 ∈ {W4..W7}

表 22-2： 指令集概述

基本指令编号	汇编助记符	汇编语法		说明	字数	周期数	影响的状态标志
1	ADD	ADD	Acc	累加器相加	1	1	OA,OB,SA,SB
		ADD	f	f = f + WREG	1	1	C,DC,N,OV,Z
		ADD	f,WREG	WREG = f + WREG	1	1	C,DC,N,OV,Z
		ADD	#lit10,Wn	Wd = lit10 + Wd	1	1	C,DC,N,OV,Z
		ADD	Wb,Ws,Wd	Wd = Wb + Ws	1	1	C,DC,N,OV,Z
		ADD	Wb,#lit5,Wd	Wd = Wb + lit5	1	1	C,DC,N,OV,Z
		ADD	Wso,#Slit4,Acc	将 16 位有符号立即数加到累加器	1	1	OA,OB,SA,SB
2	ADDC	ADDC	f	f = f + WREG + (C)	1	1	C,DC,N,OV,Z
		ADDC	f,WREG	WREG = f + WREG + (C)	1	1	C,DC,N,OV,Z
		ADDC	#lit10,Wn	Wd = lit10 + Wd + (C)	1	1	C,DC,N,OV,Z
		ADDC	Wb,Ws,Wd	Wd = Wb + Ws + (C)	1	1	C,DC,N,OV,Z
		ADDC	Wb,#lit5,Wd	Wd = Wb + lit5 + (C)	1	1	C,DC,N,OV,Z
3	AND	AND	f	f = f .AND.WREG	1	1	N,Z
		AND	f,WREG	WREG = f .AND.WREG	1	1	N,Z
		AND	#lit10,Wn	Wd = lit10 .AND.Wd	1	1	N,Z
		AND	Wb,Ws,Wd	Wd = Wb .AND.Ws	1	1	N,Z
		AND	Wb,#lit5,Wd	Wd = Wb .AND. lit5	1	1	N,Z
4	ASR	ASR	f	f = 算术右移 f	1	1	C,N,OV,Z
		ASR	f,WREG	WREG = 算术右移 f	1	1	C,N,OV,Z
		ASR	Ws,Wd	Wd = 算术右移 Ws	1	1	C,N,OV,Z
		ASR	Wb,Wns,Wnd	Wnd = 将 Wb 算术右移 Wns 位	1	1	N,Z
		ASR	Wb,#lit5,Wnd	Wnd = 将 Wb 算术右移 lit5 位	1	1	N,Z
5	BCLR	BCLR	f,#bit4	将 f 中的指定位清零	1	1	无
		BCLR	Ws,#bit4	将 Ws 中的指定位清零	1	1	无
6	BRA	BRA	C,Expr	如果有进位则转移	1	1 (2)	无
		BRA	GE,Expr	如果大于或等于则转移	1	1 (2)	无
		BRA	GEU,Expr	如果无符号大于或等于则转移	1	1 (2)	无
		BRA	GT,Expr	如果大于则转移	1	1 (2)	无
		BRA	GTU,Expr	如果无符号大于则转移	1	1 (2)	无
		BRA	LE,Expr	如果小于或等于则转移	1	1 (2)	无
		BRA	LEU,Expr	如果无符号小于或等于则转移	1	1 (2)	无
		BRA	LT,Expr	如果小于则转移	1	1 (2)	无
		BRA	LTU,Expr	如果无符号小于则转移	1	1 (2)	无
		BRA	N,Expr	如果为负则转移	1	1 (2)	无
		BRA	NC,Expr	如果没有进位则转移	1	1 (2)	无
		BRA	NN,Expr	如果不为负则转移	1	1 (2)	无
		BRA	NOV,Expr	如果未溢出则转移	1	1 (2)	无
		BRA	NZ,Expr	如果不为零则转移	1	1 (2)	无
		BRA	OA,Expr	如果累加器 A 溢出则转移	1	1 (2)	无
		BRA	OB,Expr	如果累加器 B 溢出则转移	1	1 (2)	无
		BRA	OV,Expr	如果溢出则转移	1	1 (2)	无
		BRA	SA,Expr	如果累加器 A 饱和则转移	1	1 (2)	无
		BRA	SB,Expr	如果累加器 B 饱和则转移	1	1 (2)	无
		BRA	Expr	无条件转移	1	2	无
		BRA	Z,Expr	如果为零则转移	1	1 (2)	无
		BRA	Wn	计算转移	1	2	无
7	BSET	BSET	f,#bit4	将 f 中的指定位置 1	1	1	无
		BSET	Ws,#bit4	将 Ws 中的指定位置 1	1	1	无
8	BSW	BSW.C	Ws,Wb	将 C 位内容写入 Ws<Wb>	1	1	无
		BSW.Z	Ws,Wb	将 Z 位内容写入 Ws<Wb>	1	1	无
9	BTG	BTG	f,#bit4	将 f 中的指定位翻转	1	1	无
		BTG	Ws,#bit4	将 Ws 中的指定位翻转	1	1	无

© 2007 Microchip Technology Inc.

表 22-2：　　指令集概述 （续）

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
10	BTSC	BTSC f,#bit4	对 f 中的指定位进行测试，如果为零则跳过	1	1 (2 或 3)	无
		BTSC Ws,#bit4	对 Ws 中的指定位进行测试，如果为零则跳过	1	1 (2 或 3)	无
11	BTSS	BTSS f,#bit4	对 f 中的指定位进行测试，如果为 1 则跳过	1	1 (2 或 3)	无
		BTSS Ws,#bit4	对 Ws 中的指定位进行测试，如果为 1 则跳过	1	1 (2 或 3)	无
12	BTST	BTST f,#bit4	对 f 中的指定位进行测试	1	1	Z
		BTST.C Ws,#bit4	对 Ws 中的指定位进行测试，并将其值存储到 C	1	1	C
		BTST.Z Ws,#bit4	对 Ws 中的指定位进行测试，并将其反码存储到 Z	1	1	Z
		BTST.C Ws,Wb	对 Ws<Wb> 位进行测试，并将其值存储到 C	1	1	C
		BTST.Z Ws,Wb	对 Ws<Wb> 位进行测试，并将其反码存储到 Z	1	1	Z
13	BTSTS	BTSTS f,#bit4	对 f 中的指定位进行测试，并将 f 中的该位置 1	1	1	Z
		BTSTS.C Ws,#bit4	对 Ws 中的指定位进行测试，并将其值存储到 C，然后将 Ws 中的该位置 1	1	1	C
		BTSTS.Z Ws,#bit4	对 Ws 中的指定位进行测试，并将其反码存储到 Z，然后将 Ws 中的该位置 1	1	1	Z
14	CALL	CALL lit23	调用子程序	2	2	无
		CALL Wn	间接调用子程序	1	2	无
15	CLR	CLR f	f = 0x0000	1	1	无
		CLR WREG	WREG = 0x0000	1	1	无
		CLR Ws	Ws = 0x0000	1	1	无
		CLR Acc,Wx,Wxd,Wy,Wyd,AWB	将累加器清零	1	1	OA,OB,SA,SB
16	CLRWDT	CLRWDT	将看门狗定时器清零	1	1	WDTO,Sleep
17	COM	COM f	$f = \bar{f}$	1	1	N,Z
		COM f,WREG	$WREG = \bar{f}$	1	1	N,Z
		COM Ws,Wd	$Wd = \overline{Ws}$	1	1	N,Z
18	CP	CP f	比较 f 和 WREG	1	1	C,DC,N,OV,Z
		CP Wb,#lit5	比较 Wb 和 lit5	1	1	C,DC,N,OV,Z
		CP Wb,Ws	比较 Wb 和 Ws (Wb – Ws)	1	1	C,DC,N,OV,Z
19	CP0	CP0 f	比较 f 和 0x0000	1	1	C,DC,N,OV,Z
		CP0 Ws	比较 Ws 和 0x0000	1	1	C,DC,N,OV,Z
20	CPB	CPB f	带借位比较 f 和 WREG	1	1	C,DC,N,OV,Z
		CPB Wb,#lit5	带借位比较 Wb 和 lit5	1	1	C,DC,N,OV,Z
		CPB Wb,Ws	带借位比较 Wb 和 Ws (Wb – Ws – C)	1	1	C,DC,N,OV,Z
21	CPSEQ	CPSEQ Wb, Wn	比较 Wb 和 Wn，如果相等则跳过	1	1 (2 或 3)	无
22	CPSGT	CPSGT Wb, Wn	比较 Wb 和 Wn，如果大于则跳过	1	1 (2 或 3)	无
23	CPSLT	CPSLT Wb, Wn	比较 Wb 和 Wn，如果小于则跳过	1	1 (2 或 3)	无
24	CPSNE	CPSNE Wb, Wn	比较 Wb 和 Wn，如果不相等则跳过	1	1 (2 或 3)	无
25	DAW	DAW Wn	Wn = 十进制调整 Wn	1	1	C
26	DEC	DEC f	f = f – 1	1	1	C,DC,N,OV,Z
		DEC f,WREG	WREG = f – 1	1	1	C,DC,N,OV,Z
		DEC Ws,Wd	Wd = Ws – 1	1	1	C,DC,N,OV,Z
27	DEC2	DEC2 f	f = f – 2	1	1	C,DC,N,OV,Z
		DEC2 f,WREG	WREG = f – 2	1	1	C,DC,N,OV,Z
		DEC2 Ws,Wd	Wd = Ws – 2	1	1	C,DC,N,OV,Z
28	DISI	DISI #lit14	在 k 个指令周期内禁止中断	1	1	无

表 22-2： 指令集概述 （续）

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
29	DIV	DIV.S Wm,Wn	有符号 16/16 位整数除法	1	18	N,Z,C,OV
		DIV.SD Wm,Wn	有符号 32/16 位整数除法	1	18	N,Z,C,OV
		DIV.U Wm,Wn	无符号 16/16 位整数除法	1	18	N,Z,C,OV
		DIV.UD Wm,Wn	无符号 32/16 位整数除法	1	18	N,Z,C,OV
30	DIVF	DIVF Wm,Wn	有符号 16/16 位小数除法	1	18	N,Z,C,OV
31	DO	DO #lit14,Expr	执行 DO 循环代码到 PC + Expr，执行次数为 lit14 + 1 次	2	2	无
		DO Wn,Expr	执行 DO 循环代码到 PC + Expr，执行次数为 (Wn) + 1 次	2	2	无
32	ED	ED Wm*Wm,Acc,Wx,Wy,Wxd	欧几里德距离 （无累加）	1	1	OA,OB,OAB, SA,SB,SAB
33	EDAC	EDAC Wm*Wm,Acc,Wx,Wy,Wxd	欧几里德距离	1	1	OA,OB,OAB, SA,SB,SAB
34	EXCH	EXCH Wns,Wnd	交换 Wns 和 Wnd 的内容	1	1	无
35	FBCL	FBCL Ws,Wnd	从左边 （MSb）查找第一个位变化	1	1	C
36	FF1L	FF1L Ws,Wnd	从左边 （MSb）查找第一个 1	1	1	C
37	FF1R	FF1R Ws,Wnd	从右边 （LSb）查找第一个 1	1	1	C
38	GOTO	GOTO Expr	转移到地址	2	2	无
		GOTO Wn	间接转移到地址	1	2	无
39	INC	INC f	f = f + 1	1	1	C,DC,N,OV,Z
		INC f,WREG	WREG = f + 1	1	1	C,DC,N,OV,Z
		INC Ws,Wd	Wd = Ws + 1	1	1	C,DC,N,OV,Z
40	INC2	INC2 f	f = f + 2	1	1	C,DC,N,OV,Z
		INC2 f,WREG	WREG = f + 2	1	1	C,DC,N,OV,Z
		INC2 Ws,Wd	Wd = Ws + 2	1	1	C,DC,N,OV,Z
41	IOR	IOR f	f = f .IOR.WREG	1	1	N,Z
		IOR f,WREG	WREG = f .IOR.WREG	1	1	N,Z
		IOR #lit10,Wn	Wd = lit10 .IOR.Wd	1	1	N,Z
		IOR Wb,Ws,Wd	Wd = Wb .IOR.Ws	1	1	N,Z
		IOR Wb,#lit5,Wd	Wd = Wb .IOR. lit5	1	1	N,Z
42	LAC	LAC Wso,#Slit4,Acc	装载累加器	1	1	OA,OB,OAB, SA,SB,SAB
43	LNK	LNK #lit14	分配堆栈帧	1	1	无
44	LSR	LSR f	f = 逻辑右移 f	1	1	C,N,OV,Z
		LSR f,WREG	WREG = 逻辑右移 f	1	1	C,N,OV,Z
		LSR Ws,Wd	Wd = 逻辑右移 Ws	1	1	C,N,OV,Z
		LSR Wb,Wns,Wnd	Wnd = 将 Wb 逻辑右移 Wns 位	1	1	N,Z
		LSR Wb,#lit5,Wnd	Wnd = 将 Wb 逻辑右移 lit5 位	1	1	N,Z
45	MAC	MAC Wm*Wn,Acc,Wx,Wxd,Wy,Wyd ,AWB	相乘并累加	1	1	OA,OB,OAB, SA,SB,SAB
		MAC Wm*Wm,Acc,Wx,Wxd,Wy,Wyd	平方并累加	1	1	OA,OB,OAB, SA,SB,SAB
46	MOV	MOV f,Wn	将 f 中的内容送入 Wn	1	1	无
		MOV f	将 f 中的内容送入目标寄存器	1	1	N,Z
		MOV f,WREG	将 f 中的内容送入 WREG	1	1	N,Z
		MOV #lit16,Wn	将 16 位立即数送入 Wn	1	1	无
		MOV.b #lit8,Wn	将 8 位立即数送入 Wn	1	1	无
		MOV Wn,f	将 Wn 中的内容送入 f	1	1	无
		MOV Wso,Wdo	将 Ws 中的内容送入 Wd	1	1	无
		MOV WREG,f	将 WREG 中的内容送入 f	1	1	N,Z
		MOV.D Wns,Wd	将 W(ns):W(ns + 1) 中的双字内容送入 Wd	1	2	无
		MOV.D Ws,Wnd	将 Ws 中的双字内容送入 W(nd + 1):W(nd)	1	2	无
47	MOVSAC	MOVSAC Acc,Wx,Wxd,Wy,Wyd,AWB	预取操作数并保存累加器	1	1	无

© 2007 Microchip Technology Inc.

表 22-2：　　指令集概述 （续）

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
48	MPY	MPY Wm*Wn,Acc,Wx,Wxd,Wy,Wyd	Wm 与 Wn 相乘，结果存入累加器	1	1	OA,OB,OAB, SA,SB,SAB
		MPY Wm*Wm,Acc,Wx,Wxd,Wy,Wyd	Wm 平方，结果存入累加器	1	1	OA,OB,OAB, SA,SB,SAB
49	MPY.N	MPY.N Wm*Wn,Acc,Wx,Wxd,Wy,Wyd	Wm 与 Wn 相乘并取反，结果存入累加器	1	1	无
50	MSC	MSC Wm*Wm,Acc,Wx,Wxd,Wy,Wyd, AWB	相乘再从累加器中减去	1	1	OA,OB,OAB, SA,SB,SAB
51	MUL	MUL.SS Wb,Ws,Wnd	{Wnd + 1, Wnd} = signed(Wb) * signed(Ws)	1	1	无
		MUL.SU Wb,Ws,Wnd	{Wnd + 1, Wnd} = signed(Wb) * unsigned(Ws)	1	1	无
		MUL.US Wb,Ws,Wnd	{Wnd + 1, Wnd} = unsigned(Wb) * signed(Ws)	1	1	无
		MUL.UU Wb,Ws,Wnd	{Wnd + 1, Wnd} = unsigned(Wb) * unsigned(Ws)	1	1	无
		MUL.SU Wb,#lit5,Wnd	{Wnd + 1, Wnd} = signed(Wb) * unsigned(lit5)	1	1	无
		MUL.UU Wb,#lit5,Wnd	{Wnd + 1, Wnd} = unsigned(Wb) * unsigned(lit5)	1	1	无
		MUL f	W3:W2 = f * WREG	1	1	无
52	NEG	NEG Acc	将累加器内容求补	1	1	OA,OB,OAB, SA,SB,SAB
		NEG f	f = $\bar{f}$ + 1	1	1	C,DC,N,OV,Z
		NEG f,WREG	WREG = $\bar{f}$ + 1	1	1	C,DC,N,OV,Z
		NEG Ws,Wd	Wd = $\overline{Ws}$ + 1	1	1	C,DC,N,OV,Z
53	NOP	NOP	空操作	1	1	无
		NOPR	空操作	1	1	无
54	POP	POP f	将栈顶 （TOS）的内容弹出到 f	1	1	无
		POP Wdo	将栈顶 （TOS）的内容弹出到 Wdo	1	1	无
		POP.D Wnd	将栈顶 （TOS）的内容弹出到 W(nd):W(nd +1)	1	2	无
		POP.S	将影子寄存器的内容弹出到主寄存器	1	1	全部
55	PUSH	PUSH f	将 f 的内容压入栈顶 （TOS）	1	1	无
		PUSH Wso	将 Wso 的内容压入栈顶 （TOS）	1	1	无
		PUSH.D Wns	将 W(ns):W(ns + 1) 的双字内容压入栈顶 （TOS）	1	2	无
		PUSH.S	将主寄存器中的双字内容压入影子寄存器	1	1	无
56	PWRSAV	PWRSAV #lit1	进入休眠或空闲模式	1	1	WDTO,Sleep
57	RCALL	RCALL Expr	相对调用	1	2	无
		RCALL Wn	计算调用	1	2	无
58	REPEAT	REPEAT #lit14	将下一条指令重复执行 lit14 + 1 次	1	1	无
		REPEAT Wn	将下一条指令重复执行 (Wn) + 1 次	1	1	无
59	RESET	RESET	软件器件复位	1	1	无
60	RETFIE	RETFIE	从中断返回	1	3 (2)	无
61	RETLW	RETLW #lit10,Wn	返回并将立即数存入 Wn	1	3 (2)	无
62	RETURN	RETURN	从子程序返回	1	3 (2)	无
63	RLC	RLC f	f = 对 f 执行带进位的循环左移	1	1	C,N,Z
		RLC f,WREG	WREG = 对 f 执行带进位的循环左移	1	1	C,N,Z
		RLC Ws,Wd	Wd = 对 Ws 执行带进位的循环左移	1	1	C,N,Z
64	RLNC	RLNC f	f = 循环左移 f （不带进位）	1	1	N,Z
		RLNC f,WREG	WREG = 循环左移 f （不带进位）	1	1	N,Z
		RLNC Ws,Wd	Wd = 循环左移 Ws （不带进位）	1	1	N,Z
65	RRC	RRC f	f = 对 f 执行带进位的循环右移	1	1	C,N,Z
		RRC f,WREG	WREG = 对 f 执行带进位的循环右移	1	1	C,N,Z
		RRC Ws,Wd	Wd = 对 Ws 执行带进位的循环右移	1	1	C,N,Z

表 22-2： 指令集概述 （续）

基本指令编号	汇编助记符	汇编语法		说明	字数	周期数	影响的状态标志
66	RRNC	RRNC	f	f = 循环右移 f （不带进位）	1	1	N,Z
		RRNC	f,WREG	WREG = 循环右移 f （不带进位）	1	1	N,Z
		RRNC	Ws,Wd	Wd = 循环右移 Ws （不带进位）	1	1	N,Z
67	SAC	SAC	Acc,#Slit4,Wdo	保存累加器内容	1	1	无
		SAC.R	Acc,#Slit4,Wdo	保存舍入后的累加器内容	1	1	无
68	SE	SE	Ws,Wnd	Wnd = 符号扩展后的 Ws	1	1	C,N,Z
69	SETM	SETM	f	f = 0xFFFF	1	1	无
		SETM	WREG	WREG = 0xFFFF	1	1	无
		SETM	Ws	Ws = 0xFFFF	1	1	无
70	SFTAC	SFTAC	Acc,Wn	对累加器算术移位 (Wn) 次	1	1	OA,OB,OAB, SA,SB,SAB
		SFTAC	Acc,#Slit6	对累加器算术移位 Slit6 次	1	1	OA,OB,OAB, SA,SB,SAB
71	SL	SL	f	f = 左移 f	1	1	C,N,OV,Z
		SL	f,WREG	WREG = 左移 f	1	1	C,N,OV,Z
		SL	Ws,Wd	Wd = 左移 Ws	1	1	C,N,OV,Z
		SL	Wb,Wns,Wnd	Wnd = 将 Wb 左移 Wns 位	1	1	N,Z
		SL	Wb,#lit5,Wnd	Wnd = 将 Wb 左移 lit5 位	1	1	N,Z
72	SUB	SUB	Acc	从累加器减去	1	1	OA,OB,OAB, SA,SB,SAB
		SUB	f	f = f – WREG	1	1	C,DC,N,OV,Z
		SUB	f,WREG	WREG = f – WREG	1	1	C,DC,N,OV,Z
		SUB	#lit10,Wn	Wn = Wn – lit10	1	1	C,DC,N,OV,Z
		SUB	Wb,Ws,Wd	Wd = Wb – Ws	1	1	C,DC,N,OV,Z
		SUB	Wb,#lit5,Wd	Wd = Wb – lit5	1	1	C,DC,N,OV,Z
73	SUBB	SUBB	f	f = f – WREG – ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB	f,WREG	WREG = f – WREG – ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB	#lit10,Wn	Wn = Wn – lit10 – ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB	Wb,Ws,Wd	Wd = Wb – Ws – ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB	Wb,#lit5,Wd	Wd = Wb – lit5 – ($\overline{C}$)	1	1	C,DC,N,OV,Z
74	SUBR	SUBR	f	f = WREG – f	1	1	C,DC,N,OV,Z
		SUBR	f,WREG	WREG = WREG – f	1	1	C,DC,N,OV,Z
		SUBR	Wb,Ws,Wd	Wd = Ws – Wb	1	1	C,DC,N,OV,Z
		SUBR	Wb,#lit5,Wd	Wd = lit5 – Wb	1	1	C,DC,N,OV,Z
75	SUBBR	SUBBR	f	f = WREG – f – ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBBR	f,WREG	WREG = WREG – f – ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBBR	Wb,Ws,Wd	Wd = Ws – Wb – ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBBR	Wb,#lit5,Wd	Wd = lit5 – Wb – ($\overline{C}$)	1	1	C,DC,N,OV,Z
76	SWAP	SWAP.b	Wn	Wn = 半字节交换 Wn 内容	1	1	无
		SWAP	Wn	Wn = 将 Wn 的两个字节相交换	1	1	无
77	TBLRDH	TBLRDH	Ws,Wd	将程序存储单元的 <23:16> 读入 Wd<7:0>	1	2	无
78	TBLRDL	TBLRDL	Ws,Wd	将程序存储单元的 <15:0> 读入 Wd	1	2	无
79	TBLWTH	TBLWTH	Ws,Wd	将 Ws<7:0> 写入程序存储单元的 <23:16>	1	2	无
80	TBLWTL	TBLWTL	Ws,Wd	将 Ws 写入程序存储单元的 <15:0>	1	2	无
81	ULNK	ULNK		释放堆栈帧	1	1	无
82	XOR	XOR	f	f = f .XOR.WREG	1	1	N,Z
		XOR	f,WREG	WREG = f .XOR.WREG	1	1	N,Z
		XOR	#lit10,Wn	Wd = lit10 .XOR.Wd	1	1	N,Z
		XOR	Wb,Ws,Wd	Wd = Wb .XOR.Ws	1	1	N,Z
		XOR	Wb,#lit5,Wd	Wd = Wb .XOR. lit5	1	1	N,Z
83	ZE	ZE	Ws,Wnd	Wnd = 零扩展后的 Ws	1	1	C,Z,N

© 2007 Microchip Technology Inc.

23.0 开发支持

一系列硬件及软件开发工具对 PIC® 单片机提供支持：

- 集成开发环境
  - MPLAB® IDE 软件
- 汇编器 / 编译器 / 链接器
  - MPASM™ 汇编器
  - MPLAB C18 和 MPLAB C30 C 编译器
  - MPLINK™ 目标链接器 / MPLIB™ 目标库管理器
  - MPLAB ASM30 汇编器 / 链接器 / 库
- 模拟器
  - MPLAB SIM 软件模拟器
- 仿真器
  - MPLAB ICE 2000 在线仿真器
  - MPLAB REAL ICE™ 在线仿真器
- 在线调试器
  - MPLAB ICD 2
- 器件编程器
  - PICSTART® Plus 开发编程器
  - MPLAB PM3 器件编程器
  - PICkit™ 2 开发编程器
- 低成本演示和开发板及评估工具包

23.1 MPLAB 集成开发环境软件

MPLAB IDE 软件为 8/16 位单片机市场提供了前所未有的易于使用的软件开发平台。MPLAB IDE 是基于 Windows® 操作系统的应用软件，包括：

- 一个包含所有调试工具的图形界面
  - 模拟器
  - 编程器 （单独销售）
  - 仿真器 （单独销售）
  - 在线调试器 （单独销售）
- 具有彩色上下文代码显示的全功能编辑器
- 多项目管理器
- 内容可直接编辑的可定制式数据窗口
- 高级源代码调试
- 可视化器件初始化程序，便于进行寄存器的初始化
- 鼠标停留在变量上进行查看的功能
- 通过拖放把变量从源代码窗口拉到观察窗口
- 丰富的在线帮助
- 集成了可选的第三方工具，如 HI-TECH 软件 C 编译器和 IAR C 编译器

MPLAB IDE 可以让您：

- 编辑源文件 （汇编语言或 C 语言）
- 点击一次即可完成汇编 （或编译）并将代码下载到 PIC MCU 仿真器和模拟器工具中 （自动更新所有项目信息）
- 可使用如下各项进行调试：
  - 源文件 （汇编语言或 C 语言）
  - 混合汇编语言和 C 语言
  - 机器码

MPLAB IDE 在单个开发范例中支持使用多种调试工具，包括从成本效益高的模拟器到低成本的在线调试器，再到全功能的仿真器。这样缩短了用户升级到更加灵活而功能更强大的工具时的学习时间。

23.2 MPASM 汇编器

MPASM 汇编器是全功能通用宏汇编器，适用于所有的 PIC MCU。

MPASM 汇编器可生成用于 MPLINK 目标链接器的可重定位目标文件、Intel® 标准 HEX 文件、详细描述存储器使用状况和符号参考的 MAP 文件、包含源代码行及生成机器码的绝对 LST 文件以及用于调试的 COFF 文件。

MPASM 汇编器具有如下特征：

- 集成在 MPLAB IDE 项目中
- 用户定义的宏可简化汇编代码
- 对多用途源文件进行条件汇编
- 允许完全控制汇编过程的指令

23.3 MPLAB C18 和 MPLAB C30 C 编译器

MPLAB C18 和 MPLAB C30 代码开发系统是完全的 ANSI C 编译器，分别适用于 Microchip 的 PIC18 和 PIC24 系列单片机及 dsPIC30F 和 dsPIC33 系列数字信号控制器。这些编译器可提供其他编译器并不具备的强大的集成功能和出众的代码优化能力，且使用方便。

为便于源代码调试，编译器提供了针对 MPLAB IDE 调试器的优化符号信息。

23.4 MPLINK 目标链接器 / MPLIB 目标库管理器

MPLINK 目标链接器包含了由 MPASM 汇编器、MPLAB C18 C 编译器产生的可重定位目标。通过使用链接器脚本中的指令，它还可链接预编译库中的可重定位目标。

MPLIB目标库管理器管理预编译代码库文件的创建和修改。当从源文件调用库中的一段子程序时，只有包含此子程序的模块被链接到应用中。这样可使大型库在许多不同应用中被高效地利用。

目标链接器 / 库管理器具有如下特征：

- 高效地连接单个的库而不是许多小文件
- 通过将相关的模块组合在一起来增强代码的可维护性
- 只要列出、替换、删除和抽取模块，便可灵活地创建库

23.5 MPLAB ASM30 汇编器、链接器和库管理器

MPLAB ASM30 汇编器为 dsPIC30F 器件提供转换自符号汇编语言的可重定位机器码。 MPLAB C30 C 编译器使用该汇编器生成目标文件。汇编器产生可重定位目标文件之后，可将这些目标文件存档，或与其他可重定位目标文件和存档链接以生成可执行文件。该汇编器有如下显著特征：

- 支持整个 dsPIC30F 指令集
- 支持定点数据和浮点数据
- 命令行界面
- 丰富的指令集
- 灵活的宏语言
- MPLAB IDE 兼容性

23.6 MPLAB SIM 软件模拟器

MPLAB SIM软件模拟器在指令级对PIC MCU和dsPIC® DSC 进行模拟，使得用户可以在 PC 主机的环境下进行代码开发。对于任何给定的指令，用户均可对数据区进行检查或修改，并通过各种触发机制来产生激励。可以将各寄存器的情况记录在文件中，以便进行进一步地运行时分析。跟踪缓冲器和逻辑分析器的显示使模拟器还能记录和跟踪程序的执行、 I/O 的动作、大部分的外设及内部寄存器的状况。

MPLAB SIM 软件模拟器完全支持使用 MPLAB C18 和 MPLAB C30 C 编译器以及MPASM和MAPLAB ASM30 汇编器的符号调试。该软件模拟器可用于在硬件实验室环境外灵活地开发和调试代码，是一款完美且经济的软件开发工具。

© 2007 Microchip Technology Inc.

23.7 MPLAB ICE 2000 高性能在线仿真器

MPLAB ICE 2000 在线仿真器旨在为产品开发工程师提供一整套用于 PIC 单片机的设计工具。 MPLAB ICE 2000 在线仿真器的软件控制由 MPLAB 集成开发环境平台提供，它允许在单一环境下进行编辑、编译、下载以及源代码调试。

MPLAB ICE 2000 是全功能仿真器系统，它具有增强的跟踪、触发和数据监控功能。处理器模块可插拔，使系统可轻松进行重新配置以适应各种不同处理器的仿真需要。MPLAB ICE 2000 在线仿真器的架构允许对其进行扩展以支持新的 PIC 单片机。

MPLAB ICE 2000 在线仿真器系统设计为一款实时仿真系统，该仿真系统具备通常只有昂贵的开发工具中才有的高级功能。选择 PC 平台和 Microsoft® Windows® 32 位操作系统可使这些功能在一个简单而统一的应用中得到很好的利用。

23.8 MPLAB REAL ICE 在线仿真器系统

MPLAB REAL ICE在线仿真器系统是Microchip针对其闪存 DSC 和 MCU 器件而推出的新一代高速仿真器。结合 MPLAB 集成开发环境（IDE）所具有的易于使用且功能强大的图形用户界面，该仿真器可对 PIC® 闪存 MCU 和 dsPIC® DSC 进行调试和编程。IDE 是随每个工具包一起提供的。

MPLAB REAL ICE 探针通过高速 USB 2.0 接口与设计工程师的 PC 相连，并利用与常用 MPLAB ICD 2 系统兼容的连接器 （RJ11）或新型抗噪声、高速低压差分信号（LVDS）互连电缆 （CAT5）与目标板相连。

可通过 MPLAB IDE 下载将来版本的固件，对 MPLAB REAL ICE 进行现场升级。在即将推出的 MPLAB IDE 版本中，会支持许多新器件，还将增加一些新特性，如软件断点和汇编代码跟踪等。在同类仿真器中， MPLAB REAL ICE 的优势十分明显：低成本、高速仿真、实时变量监视、跟踪分析、复杂断点、耐用的探针接口及较长（长达 3 米）的互连电缆。

23.9 MPLAB ICD 2 在线调试器

Microchip 的在线调试器 MPLAB ICD 2 是一款功能强大而成本低廉的运行时开发工具，通过 RS-232 或高速 USB 接口与 PC 主机相连。该工具基于闪存 PIC MCU，可用于开发本系列及其他 PIC MCU 和 dsPIC DSC。MPLAB ICD 2 使用了闪存器件中内建的在线调试功能。该功能结合 Microchip 的在线串行编程（In-Circuit Serial Programming™， ICSP™）协议，可在 MPLAB 集成开发环境的图形用户界面上提供成本效益很高的在线闪存调试。这使设计人员可通过设置断点、单步运行以及对变量、CPU 状态以及外设寄存器进行监视的方法实现源代码的开发和调试。其全速运行特性可对硬件和应用进行实时测试。 MPLAB ICD 2 还可用作某些 PIC 器件的开发编程器。

23.10 MPLAB PM3 器件编程器

MPLAB PM3 器件编程器是一款通用的、符合 CE 规范的器件编程器，其可编程电压设置在 VDDMIN 和 VDDMAX 之间时可靠性最高。它有一个用来显示菜单和错误信息的大 LCD 显示器（128 x 64），以及一个支持各种封装类型的可拆卸模块化插槽装置。编程器标准配置中带有一根 ICSP™ 电缆。在单机模式下， MPLAB PM3 器件编程器不必与 PC 相连即可对 PIC 器件进行读取、验证和编程。在该模式下它还可设置代码保护。 MPLAB PM3 通过 RS-232 或 USB 电缆连接到 PC 主机上。MPLAB PM3 具备高速通信能力以及优化算法，可对存储器很大的器件进行快速编程，它还采用 SD/MMC 卡用作文件存储及数据安全应用。

23.11 PICSTART Plus 开发编程器

PICSTART Plus开发编程器是一款易于使用而成本低廉的原型编程器。它通过 COM（RS-232）端口与 PC 相连。 MPLAB 集成开发环境软件使得该编程器的使用简便、高效。PICSTART Plus 开发编程器支持采用 DIP 封装的大部分 PIC 器件，其引脚数最多可达 40 个。引脚数更多的器件，如 PIC16C92X 和 PIC17C76X，可通过连接一个转接插槽来获得支持。PICSTART Plus 开发编程器符合 CE 规范。

23.12 PICkit 2 开发编程器

PICkit™ 2 开发编程器是一个低成本编程器；对于某些选定闪存器件，它也是一个调试器，通过其易于使用的接口可对众多 Microchip 的低档、中档和 PIC18F 系列闪存单片机进行编程。 PICkit 2 入门工具包中包含一个有实验布线区的开发板、十二堂系列课程、软件和 HI-TECH 的 PICC™ Lite C 编译器，有助于用户快速掌握 PIC® 单片机的使用。这一工具包为使用 Microchip 功能强大的中档闪存系列单片机进行编程、评估和应用开发，提供了所需的一切。

23.13 演示、开发和评估板

有许多演示、开发和评估板可用于各种 PIC MCU 和 dsPIC DSC，实现对全功能系统的快速应用开发。大多数的演示、开发和评估板都有实验布线区，供用户添加定制电路；还有应用固件和源代码，用于测试和修改。

这些板支持多种功能部件，包括 LED、温度传感器、开关、扬声器、 RS-232 接口、 LCD 显示器、电位计和附加 EEPROM 存储器。

演示和开发板可用于教学环境，在实验布线区设计定制电路，从而掌握各种单片机应用。

除了 PICDEM™ 和 dsPICDEM™ 演示 / 开发板系列电路外， Microchip 还有一系列评估工具包和演示软件，适用于模拟滤波器设计、 KEELOQ® 数据安全产品 IC、CAN、IrDA®、PowerSmart 电池管理、SEEVAL® 评估系统、 Σ–Δ ADC、流速传感器，等等。

有关演示、开发和评估工具包的完整列表，请查阅 Microchip 公司网页 （www.microchip.com）。

© 2007 Microchip Technology Inc.

24.0 电气特性

本章将对 dsPIC33FJXXXGPX06/X08/X10 电气特性进行概括介绍。其余信息将在该文档的后续版本中给出。

下面列出了 dsPIC33FJXXXGPX06/X08/X10 系列器件的绝对最大额定值。器件长时间工作在最大额定值条件下可能会影响其可靠性。我们不建议使器件在或超过本规范指定的最大额定值条件下运行。

绝对最大额定值（注 1）

环境温度 ..... -40°C 至 +85°C
储存温度 ..... -65°C 至 +150°C
V_{DD} 引脚相对于 V_{SS} 的电压 ..... -0.3V 至 +4.0V
任一模拟 / 数字引脚和 $\overline{MCLR}$ 引脚相对于 V_{SS} 的电压 ..... -0.3V 至 （V_{DD} + 0.3V）
任一只能用作数字的引脚相对于 V_{SS} 的电压 ..... -0.3V 至 +5.6V
V_{DDCORE} 引脚相对于 V_{SS} 的电压 ..... 2.25V 至 2.75V
流出 V_{SS} 引脚的最大电流 ..... 300 mA
流入 V_{DD} 引脚的最大电流 （**注 2**） ..... 250 mA
任一 I/O 引脚的最大输出灌电流 （**注 3**） ..... 4 mA
任一 I/O 引脚的最大输出拉电流 （**注 3**） ..... 4 mA
所有端口的最大灌电流 ..... 200 mA
所有端口的最大拉电流 （**注 2**） ..... 200 mA

注 1： 如果器件工作条件超过上述 “绝对最大额定值”，可能引起器件永久性损坏。这仅是极限参数，我们不建议器件工作在极限值甚至超过上述极限值。器件长时间工作在极限条件下可能会影响其可靠性。

2： 允许的最大电流由器件最大功耗决定 （见表 24-2）。

3： CLKOUT 引脚例外，其灌 / 拉电流为 25 mA，另外 V_{REF+}、V_{REF-}、SCLx、SDAx、PGCx 和 PGDx 引脚的灌 / 拉电流为 12 mA。

24.1 直流特性

表 24-1： 工作 MIPS 与电压

特性	VDD 范围（单位：V）	温度范围（单位：°C）	最大 MIPS
			dsPIC33FJXXXGPX06/X08/X10
DC5	3.0-3.6V	-40°C 至 +85°C	40

表 24-2： 热工作条件

参数	符号	最小值	典型值	最大值	单位
dsPIC33FJXXXGPX06/X08/X10					
工作结温范围	TJ	-40	—	+125	°C
工作环境温度范围	TA	-40	—	+85	°C
功耗： 芯片内部功耗： PINT = VDD x (IDD – Σ IOH) I/O 引脚功耗： I/O = Σ ({VDD – VOH} x IOH) + Σ (VOL x IOL)	PD	PINT + PI/O			W
最大允许功耗	PDMAX	(TJ – TA)/θJA			W

表 24-3： 热封装特性

特性	符号	典型值	最大值	单位	注
封装热阻， 100 引脚 TQFP （14x14x1 mm）	θJA	48.4	—	°C/W	1
封装热阻， 100 引脚 TQFP （12x12x1 mm）	θJA	52.3	—	°C/W	1
封装热阻， 80 引脚 TQFP （12x12x1 mm）	θJA	38.7	—	°C/W	1
封装热阻， 64 引脚 TQFP （10x10x1 mm）	θJA	38.3	—	°C/W	1

注 1： 通过封装模拟获得结点与环境的热阻值 θJA。

表 24-4： 直流温度和电压规范

直流特性			标准工作条件：3.0V 至 3.6V（除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级）				
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
工作电压							
DC10	供电电压						
	VDD		3.0	—	3.6	V	
DC12	VDR	RAM 数据保持电压 (2)	1.1	1.3	1.8	V	
DC16	VPOR	VDD 启动电压 (4) （确保内部上电复位信号）	—	—	VSS	V	
DC17	SVDD	VDD 上升速率 （确保内部上电复位信号）	0.03	—	—	V/ms	0-3.0V/0.1s
DC18	VCORE	VDD 内核 (3) 内部稳压器	2.25	—	2.75	V	电压取决于负载、温度和 VDD

注 1： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

2： 这是在不丢失 RAM 数据的前提下， VDD 的下限值。

3： 这些参数为特性值，但生产时未经测试。

4： VDD 内核电压必须维持在 VSS 至少 200 µs 以确保 POR。

© 2007 Microchip Technology Inc.

表 24-5：　　直流特性：工作电流 （IDD）

直流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度　　-40°C ≤ TA ≤ +85°C （工业级）			
参数编号	**典型值 (1)**	**最大值**	**单位**	**条件**		
工作电流 （IDD）(2)						
DC20d	24	29	mA	-40°C	3.3V	10 MIPS
DC20	27	30	mA	+25°C		
DC20a	27	31	mA	+85°C		
DC21d	36	42	mA	-40°C	3.3V	16 MIPS
DC21	37	42	mA	+25°C		
DC21a	38	43	mA	+85°C		
DC22d	43	50	mA	-40°C	3.3V	20 MIPS
DC22	46	51	mA	+25°C		
DC22a	46	52	mA	+85°C		
DC23d	61	70	mA	-40°C	3.3V	30 MIPS
DC23	65	70	mA	+25°C		
DC23a	65	71	mA	+85°C		
DC24d	83	88	mA	-40°C	3.3V	40 MIPS
DC24	84	88	mA	+25°C		
DC24a	84	89	mA	+85°C		

注　1： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

2： 供电电流主要受工作电压和频率的影响。其他因素如 I/O 引脚负载和开关速率、振荡器类型、内部代码执行模式以及温度也对电流消耗有影响。所有 IDD 测量的测试条件为：OSC1 使用满幅的外部方波进行驱动。所有 I/O 引脚配置为输入且被拉到 VSS。MCLR = VDD，WDT 和 FSCM 被禁止。CPU、SRAM、程序存储器和数据存储器处于工作状态。外设模块不工作；但是，仍然为每个外设提供时钟 （PMD 的所有位均为零）。

表 24-6： 直流特性：空闲电流 （IIDLE）

直流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级）			
参数编号	典型值 (1)	最大值	单位	条件		
空闲电流 （IIDLE）：内核不工作、时钟工作时的基本电流 (2)						
DC40d	3	7	mA	-40°C	3.3V	10 MIPS
DC40	3	7	mA	+25°C		
DC40a	3	8	mA	+85°C		
DC40d	5	10	mA	-40°C	3.3V	16 MIPS
DC41	5	10	mA	+25°C		
DC41a	6	11	mA	+85°C		
DC42d	9	12	mA	-40°C	3.3V	20 MIPS
DC42	9	15	mA	+25°C		
DC42a	10	16	mA	+85°C		
DC43d	15	17	mA	-40°C	3.3V	30 MIPS
DC43	15	21	mA	+25°C		
DC43a	15	22	mA	+85°C		
DC44d	16	21	mA	-40°C	3.3V	40 MIPS
DC44	16	23	mA	+25°C		
DC44a	16	24	mA	+85°C		

注 1： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

2： 基本 IIDLE 电流的测量是在内核不工作、时钟工作而所有外设模块关闭的条件下进行的。外设模块禁止 SFR 寄存器为全零。所有 I/O 引脚配置为输入且被拉到 VSS。

表 24-7： 直流特性：掉电电流 （IPD）

直流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级）			
参数编号	典型值 (1)	最大值	单位	条件		
掉电电流 （IPD） (2)						
DC60d	290	963	µA	-40°C	3.0V	基本掉电电流 (3,4)
DC60	293	988	µA	+25°C		
DC60a	317	990	µA	+85°C		
DC61d	8	13	µA	-40°C	3.0V	看门狗定时器电流：ΔIWDT (3)
DC61	10	15	µA	+25°C		
DC61a	12	20	µA	+85°C		

注 1： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

2： 基本 IPD 是在所有外设和时钟都关闭的条件下进行测量的。所有 I/O 引脚配置为输入且被拉到 VSS。 WDT 等外设也都被关闭。

3： Δ 电流为当模块使能时额外消耗的电流。此电流应被加到基本 IPD 电流。

4： 这些电流是针对该系列中存储容量最大的器件测得的。

© 2007 Microchip Technology Inc.

表 24-8： 直流特性：打盹电流 （IDOZE）

直流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级）		
参数编号	典型值 (1)	最大值	打盹模式 时钟分频比	单位	条件
DC73a	25	32	1:2	mA -40°C	3.3V 40 MIPS
DC73f	23	27	1:64		
DC73g	23	26	1:128		
DC70a	42	47	1:2	mA +25°C	
DC70f	26	27	1:64		
DC70g	25	27	1:128		
DC71a	41	48	1:2	mA +85°C	
DC71f	25	28	1:64		
DC71g	24	28	1:128		

注 1： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

表 24-9：　直流特性：I/O 引脚输入规范

直流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度　-40°C ≤ TA ≤ +85°C （工业级）				
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
	VIL	**输入低电压**					
DI10		I/O 引脚	VSS	—	0.2 VDD	V	
DI15		$\overline{\text{MCLR}}$	VSS	—	0.2 VDD	V	
DI16		OSC1 （XT 模式）	VSS	—	0.2 VDD	V	
DI17		OSC1 （HS 模式）	VSS	—	0.2 VDD	V	
DI18		SDAx 和 SCLx	VSS	—	0.3 VDD	V	禁止 SMbus
DI19		SDAx 和 SCLx	VSS	—	0.2 VDD	V	使能 SMbus
	VIH	**输入高电压**					
DI20		I/O 引脚： 带模拟功能 仅数字功能	 0.8 VDD 0.8 VDD	 — —	 VDD 5.5	 V V	
DI25		$\overline{\text{MCLR}}$	0.8 VDD	—	VDD	V	
DI26		OSC1 （XT 模式）	0.7 VDD	—	VDD	V	
DI27		OSC1 （HS 模式）	0.7 VDD	—	VDD	V	
DI28		SDAx 和 SCLx	0.7 VDD	—	VDD	V	禁止 SMbus
DI29		SDAx 和 SCLx	0.8 VDD	—	VDD	V	使能 SMbus
	ICNPU	**CNx 上拉电流**					
DI30			50	250	400	μA	VDD = 3.3V， VPIN = VSS
	IIL	**输入泄漏电流 (2,3)**					
DI50		I/O 端口	—	—	±2	μA	VSS ≤ VPIN ≤ VDD， 引脚处于高阻态
DI51		模拟输入引脚	—	—	±1	μA	VSS ≤ VPIN ≤ VDD， 引脚处于高阻态
DI51A		模拟输入引脚	—	—	±2	μA	模拟引脚与外部参考引脚共用
DI55		$\overline{\text{MCLR}}$	—	—	±2	μA	VSS ≤ VPIN ≤ VDD
DI56		OSC1	—	—	±2	μA	VSS ≤ VPIN ≤ VDD， XT 和 HS 模式

注　1： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

2： $\overline{\text{MCLR}}$ 引脚上的泄漏电流主要取决于所施加电压。规定电压为正常工作条件下的电压。在不同的输入电压下可能测得更高的泄漏电流。

3： 负电流定义为引脚的拉电流。

© 2007 Microchip Technology Inc.

表 24-10： 直流特性：I/O 引脚输出规范

直流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级）				
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
	VOL	输出低电压					
DO10		I/O 端口	—	—	0.4	V	IOL = 2 mA， VDD = 3.3V
DO16		OSC2/CLKO	—	—	0.4	V	IOL = 2 mA， VDD = 3.3V
	VOH	输出高电压					
DO20		I/O 端口	2.40	—	—	V	IOH = -2.3 mA， VDD = 3.3V
DO26		OSC2/CLKO	2.41	—	—	V	IOH = -1.3 mA， VDD = 3.3V

表 24-11： 电气特性：BOR

直流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级）				
参数编号	符号	特性	最小值 (1)	典型值	最大值 (1)	单位	条件
BO10	VBOR	当 VDD 从高电压翻转至低电压时的 BOR 事件 BOR 事件与 VDD 内核电压下降关联	2.40	—	2.55	V	-40°C 至 +85°C

注 1： 这些参数仅供设计参考，生产时未经测试。

表 24-12： 直流特性：程序存储器

直流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级）				
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
		闪存程序存储器					
D130	EP	单元耐擦写能力	100	1000	—	E/W	-40°C 至 +85°C
D131	VPR	读操作时的 VDD	VMIN	—	3.6	V	VMIN = 最小工作电压
D132B	VPEW	自定时写的 VDD	VMIN	—	3.6	V	VMIN = 最小工作电压
D134	TRETD	特性保持时间	20	—	—	年	假设没有违反其他规范
D135	IDDP	编程时的供电电流	—	10	—	mA	
D136	TRW	行写入时间	—	1.6	—	ms	
D137	TPE	页擦除时间	—	20	—	ms	
D138	TWW	字写周期	20	—	40	µs	

注 1： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

表 24-13： 内部稳压器规范

工作条件：-40°C < TA < +85°C （除非另外声明）							
参数编号	符号	特性	最小值	典型值	最大值	单位	备注
	CEFC	外部滤波电容值	1	10	—	µF	电容必须与一个低阻值的电阻 （< 5 欧姆）串联

© 2007 Microchip Technology Inc.

24.2 交流特性和时序参数

本节包含的信息说明了dsPIC33FJXXXGPX06/X08/X10
系列器件的交流特性和时序参数。

表 24-14： 温度和电压规范——交流

交流特性	**标准工作条件：3.0V 至 3.6V** **（除非另外声明）** 工作温度 -40°C ≤ TA ≤ +85°C （工业级） 工作电压 VDD 范围如**第 24.0 节 “电气特性”**中所介绍。

图 24-1： 器件时序规范的负载条件

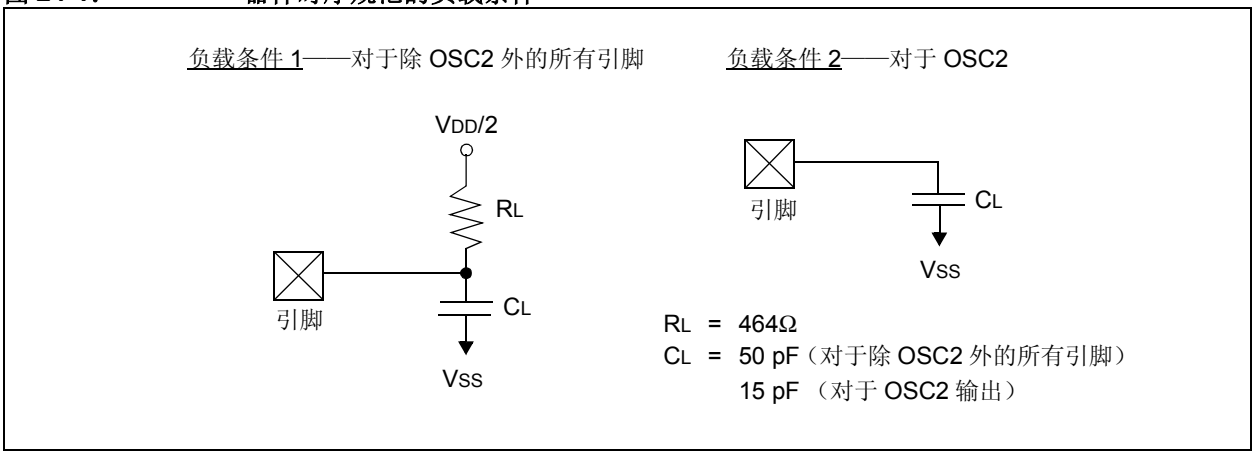


表 24-15： 输出引脚上的容性负载要求

参数编号	符号	特性	最小值	典型值	最大值	单位	条件
DO50	COSC2	OSC2/SOSC2 引脚	—	—	15	pF	当外部时钟用于驱动 OSC1 时处于 XT 和 HS 模式下
DO56	CIO	所有 I/O 引脚和 OSC2	—	—	50	pF	EC 模式
DO58	CB	SCLx 和 SDAx	—	—	400	pF	在 I^2C™ 模式下

图 24-2：　　外部时钟时序

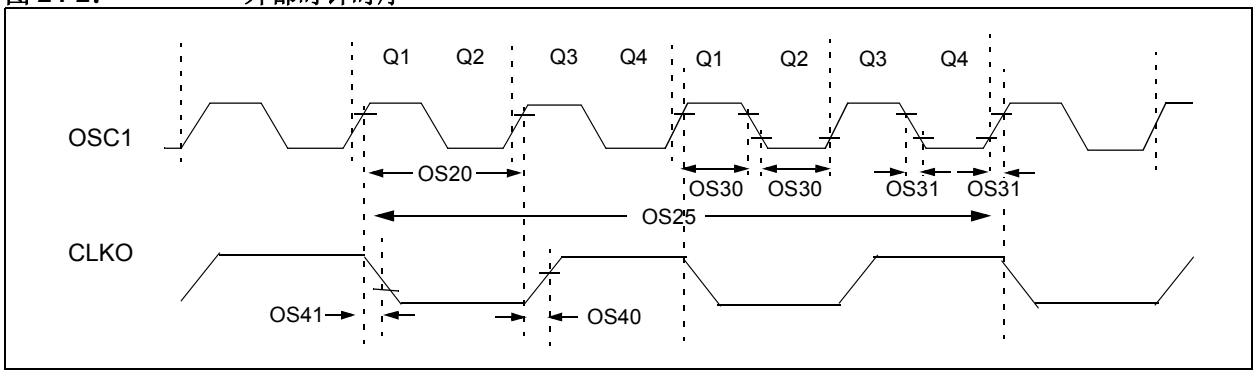


表 24-16：　　外部时钟时序要求

交流特性			**标准工作条件：3.0V 至 3.6V（除非另外声明）** 工作温度　　-40°C ≤ TA ≤ +85°C （工业级）				
参数编号	**符号**	**特性**	**最小值**	**典型值 (1)**	**最大值**	**单位**	**条件**
OS10	FIN	外部 CLKI 频率 （外部时钟仅允许运行于 EC 和 ECPLL 模式）	DC	—	40	MHz	EC
		晶振频率	3.5 10 —	— — —	10 40 33	MHz MHz kHz	XT HS SOSC
OS20	TOSC	TOSC = 1/FOSC	12.5	—	DC	ns	
OS25	TCY	指令周期 (2)	25	—	DC	ns	
OS30	TosL，TosH	外部时钟输入 （OSC1） 高电平或低电平时间	0.375 x TOSC	—	0.625 x TOSC	ns	EC
OS31	TosR，TosF	外部时钟输入 （OSC1） 上升或下降时间	—	—	20	ns	EC
OS40	TckR	CLKO 上升时间 (3)	—	5.2	—	ns	
OS41	TckF	CLKO 下降时间 (3)	—	5.2	—	ns	

注　1： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

2： 指令周期 （TCY）等于输入振荡器时基周期的两倍。所有规定值均为基于针对特定振荡器类型，器件在标准工作条件下执行代码时的特性数据。超出这些规定的限定值，可能导致振荡器运行不稳定和 / 或导致电流消耗超出预期值。所有器件在测试 “最小”值时，都在 OSC1/CLKI 引脚连接了外部时钟。当使用了外部时钟输入时，所有器件的 “最大”周期时间限制为 “DC”（无时钟）。

3： 测量在 EC 模式下进行。在 OSC2 引脚上测量 CLKO 信号。

© 2007 Microchip Technology Inc.

表 24-17： PLL 时钟时序规范 （VDD = 3.0V 至 3.6V）

交流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级）				
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
OS50	FPLLI	PLL 压控振荡器 （VCO）的输入频率范围 (2)	0.8	—	8.0	MHz	ECPLL、 HSPLL 和 XTPLL 模式
OS51	FSYS	片上 VCO 系统频率	100	—	200	MHz	
OS52	TLOCK	PLL 起振时间 （锁定时间）	0.9	1.5	3.1	ms	
OS53	DCLK	CLKO 稳定性 （抗抖动性）	-3.0	0.5	3.0	%	在 100 ms 时间段内测量

注 1： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

表 24-18： 交流特性：内部 RC 精度

交流特性		标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级）					
参数编号	特性	最小值	典型值	最大值	单位	条件	
	FRC 频率 = 7.37 MHz 时的内部 FRC 精度 (1,2)						
F20	FRC	-2	—	+2	%	-40°C ≤ TA ≤ +85°C	VDD = 3.0-3.6V

注 1： 频率在 25°C 和 3.3V 条件下校准。 TUN 位可用来补偿温度漂移。

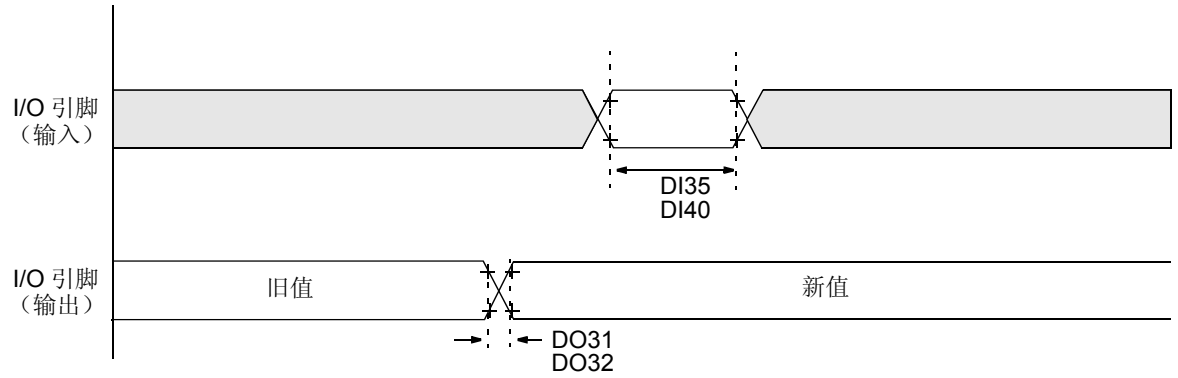
2： FRC 被设置为 25°C FRC 下的初始频率 7.37 MHz （±2%）。

表 24-19： 内部 RC 精度

交流特性		标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级）					
参数编号	特性	最小值	典型值	最大值	单位	条件	
	频率为 32.768 kHz 时的 LPRC(1)						
F21		-20	±6	+20	%	-40°C ≤ TA ≤ +85°C	VDD = 3.0-3.6V

注 1： LPRC 频率将随 VDD 的变化而变化。

图 24-3： CLKO 和 I/O 时序特性



注：负载条件请参见图 24-1。

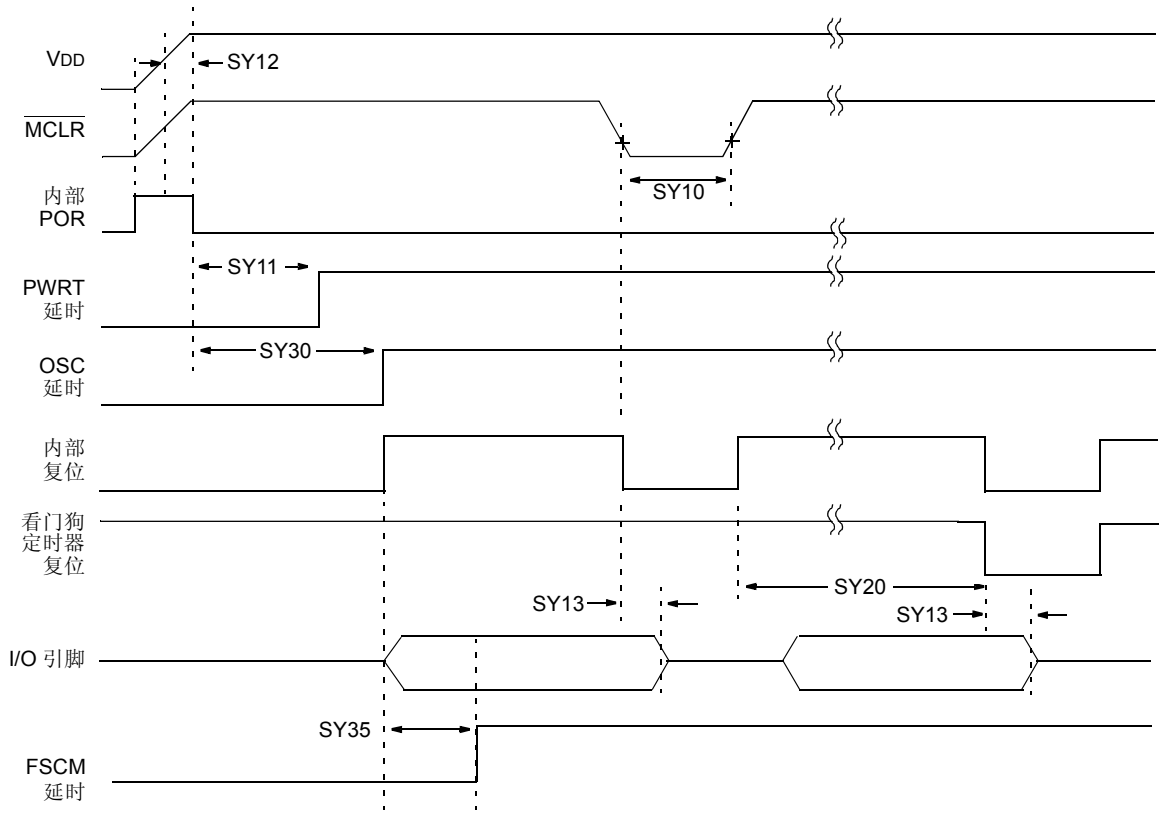
表 24-20： CLKO 和 I/O 时序要求

交流特性			**标准工作条件：3.0V 至 3.6V（除非另外声明）** 工作温度 -40°C ≤ TA ≤ +85°C （工业级）				
参数编号	**符号**	**特性**	**最小值**	**典型值 (1)**	**最大值**	**单位**	**条件**
DO31	TIOR	端口输出上升时间	—	10	25	ns	—
DO32	TIOF	端口输出下降时间	—	10	25	ns	—
DI35	TINP	INTx 引脚高电平或低电平时间 （输出）	20	—	—	ns	—
DI40	TRBP	CNx 高电平或低电平时间 （输入）	2	—	—	TCY	—

注 1： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

© 2007 Microchip Technology Inc.

图 24-4：　　复位、看门狗定时器、振荡器起振定时器和上电延时定时器时序特性



注：负载条件请参见图 24-1。

表 24-21： 复位、看门狗定时器、振荡器起振定时器和上电延时定时器时序要求

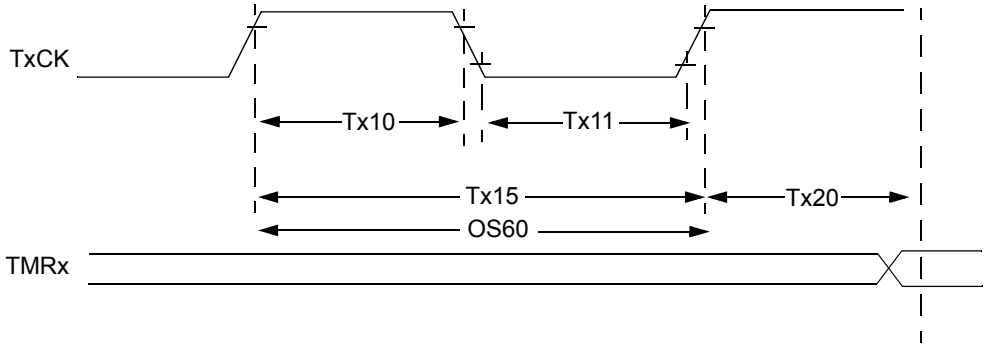
交流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SY10	TMCL	$\overline{\text{MCLR}}$ 脉冲宽度 （低电平）	2	—	—	μs	-40°C 至 +85°C
SY11	TPWRT	上电延时定时器周期	— — — — — — —	2 4 8 16 32 64 128	— — — — — — —	ms	-40°C 至 +85°C 用户可编程
SY12	TPOR	上电复位延时	3	10	30	μs	-40°C 至 +85°C
SY13	TIOZ	自 $\overline{\text{MCLR}}$ 低电平或看门狗定时器复位起 I/O 处于高阻态的时间	0.68	0.72	1.2	μs	
SY20	TWDT1	看门狗定时器超时周期 （无预分频器）	1.7	2.1	2.6	ms	VDD = 3V， -40°C 至 +85°C
SY30	TOST	振荡器起振定时器周期	—	1024 TOSC	—	—	TOSC = OSC1 周期
SY35	TFSCM	故障保护时钟监视器延时	—	500	900	μs	-40°C 至 +85°C

注 1： 这些参数为特性值，但生产时未经测试。

2： 除非另外声明，否则 “典型值”栏中的数据均为 5V 和 25°C 条件下的值。

© 2007 Microchip Technology Inc.

图 24-5：　　TIMER1, 2, 3, 4, 5, 6, 7, 8 和 9 外部时钟时序特性



注：负载条件请参见图 24-1。

表 24-22：　　TIMER1 外部时钟时序要求 (1)

交流特性				**标准工作条件：3.0V 至 3.6V** **（除非另外声明）** 工作温度　　-40°C ≤ TA ≤ +85°C				
参数编号	**符号**	**特性**		**最小值**	**典型值**	**最大值**	**单位**	**条件**
TA10	TTxH	TxCK 高电平时间	同步，无预分频器	0.5 TCY + 20	—	—	ns	也必须满足参数 TA15
			同步，带预分频器	10	—	—	ns	
			异步	10	—	—	ns	
TA11	TTxL	TxCK 低电平时间	同步，无预分频器	0.5 TCY + 20	—	—	ns	也必须满足参数 TA15
			同步，带预分频器	10	—	—	ns	
			异步	10	—	—	ns	
TA15	TTxP	TxCK 输入周期	同步，无预分频器	TCY + 40	—	—	ns	
			同步，带预分频器	取如下二者中较大值：20 ns 或 (TCY + 40)/N	—	—	—	N = 预分频值（1, 8, 64, 256）
			异步	20	—	—	ns	
OS60	Ft1	SOSC1/T1CK 振荡器输入频率范围 （通过置 1 TCS （T1CON<1>）使能振荡器）		DC	—	50	kHz	
TA20	TCKEXTMRL	从外部 TxCK 时钟边沿到定时器递增之间的延时		0.5 TCY	—	1.5 TCY	—	

注　1： Timer1 属于 A 类定时器。

表 24-23： TIMER2、 TIMER4、 TIMER6 和 TIMER8 外部时钟时序要求

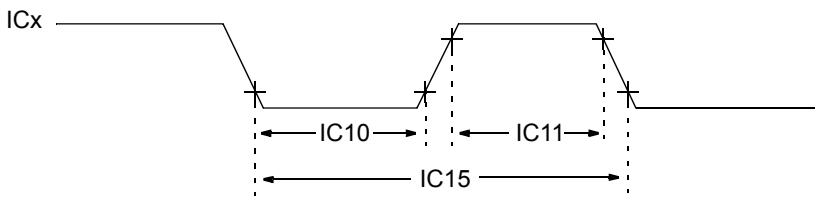
交流特性				标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C				
参数编号	符号	特性		最小值	典型值	最大值	单位	条件
TB10	TtxH	TxCK 高电平时间	同步，无预分频器	0.5 TCY + 20	—	—	ns	也必须满足参数 TB15
			同步，带预分频器	10	—	—	ns	
TB11	TtxL	TxCK 低电平时间	同步，无预分频器	0.5 TCY + 20	—	—	ns	也必须满足参数 TB15
			同步，带预分频器	10	—	—	ns	
TB15	TtxP	TxCK 输入周期	同步，无预分频器	TCY + 40	—	—	ns	N = 预分频值（1, 8, 64, 256）
			同步，带预分频器	取如下二者中较大值：20 ns 或 (TCY + 40)/N				
TB20	TCKEXTMRL	从外部 TxCK 时钟边沿到定时器递增之间的延时		0.5 TCY	—	1.5 TCY	—	

表 24-24： TIMER3、 TIMER5、 TIMER7 和 TIMER9 外部时钟时序要求

交流特性				标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C				
参数编号	符号	特性		最小值	典型值	最大值	单位	条件
TC10	TtxH	TxCK 高电平时间	同步	0.5 TCY + 20	—	—	ns	也必须满足参数 TC15
TC11	TtxL	TxCK 低电平时间	同步	0.5 TCY + 20	—	—	ns	也必须满足参数 TC15
TC15	TtxP	TxCK 输入周期	同步，无预分频器	TCY + 40	—	—	ns	N = 预分频值（1, 8, 64, 256）
			同步，带预分频器	取如下二者中较大值：20 ns 或 (TCY + 40)/N				
TC20	TCKEXTMRL	从外部 TxCK 时钟边沿到定时器递增之间的延时		0.5 TCY	—	1.5 TCY	—	

© 2007 Microchip Technology Inc.

图 24-6：　　输入捕捉 （CAPx）时序特性



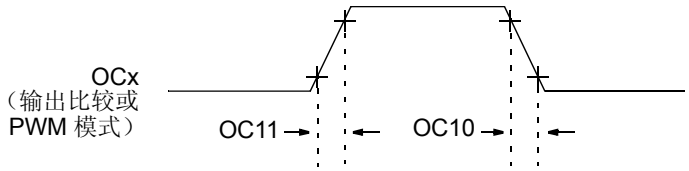
注：负载条件请参见图 24-1。

表 24-25：　　输入捕捉时序要求

交流特性			**标准工作条件：3.0V 至 3.6V** **（除非另外声明）** 工作温度　　-40°C ≤ TA ≤ +85°C				
参数编号	**符号**	**特性 (1)**		**最小值**	**最大值**	**单位**	**条件**
IC10	TccL	ICx 输入低电平时间	无预分频器	0.5 TCY + 20	—	ns	
			带预分频器	10	—	ns	
IC11	TccH	ICx 输入高电平时间	无预分频器	0.5 TCY + 20	—	ns	
			带预分频器	10	—	ns	
IC15	TccP	ICx 输入周期		(TCY + 40)/N	—	ns	N = 预分频值 （1、 4 或 16）

注　1： 这些参数为特性值，但生产时未经测试。

图 24-7：　　输出比较模块 （OCx）时序特性



注：负载条件请参见图 24-1。

表 24-26：　　输出比较模块时序要求

交流特性			**标准工作条件：3.0V 至 3.6V** **（除非另外声明）** 工作温度　　-40°C ≤ TA ≤ +85°C				
参数编号	**符号**	**特性 (1)**	**最小值**	**典型值**	**最大值**	**单位**	**条件**
OC10	TccF	OCx 输出下降时间	—	—	—	ns	见参数 D032
OC11	TccR	OCx 输出上升时间	—	—	—	ns	见参数 D031

注　1： 这些参数为特性值，但生产时未经测试。

图 24-8：　　输出比较 /PWM 模块时序特性

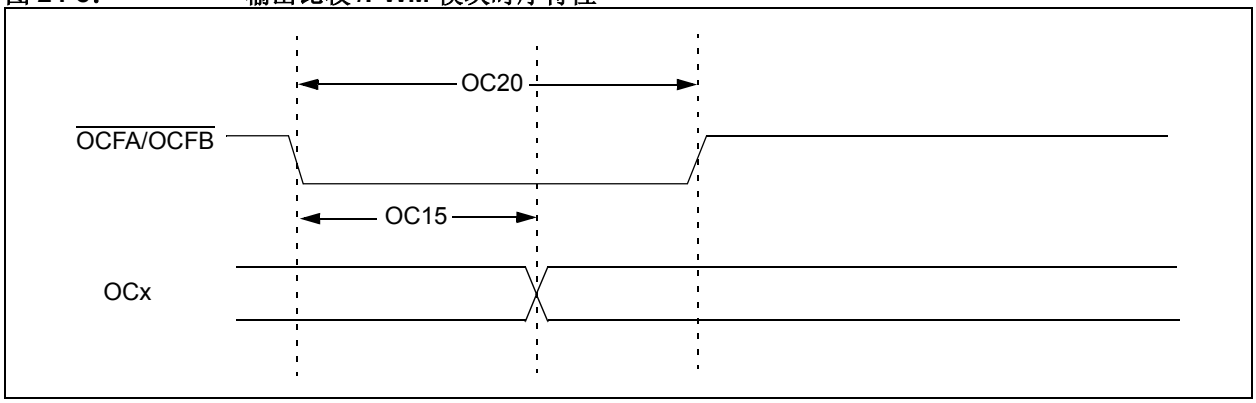


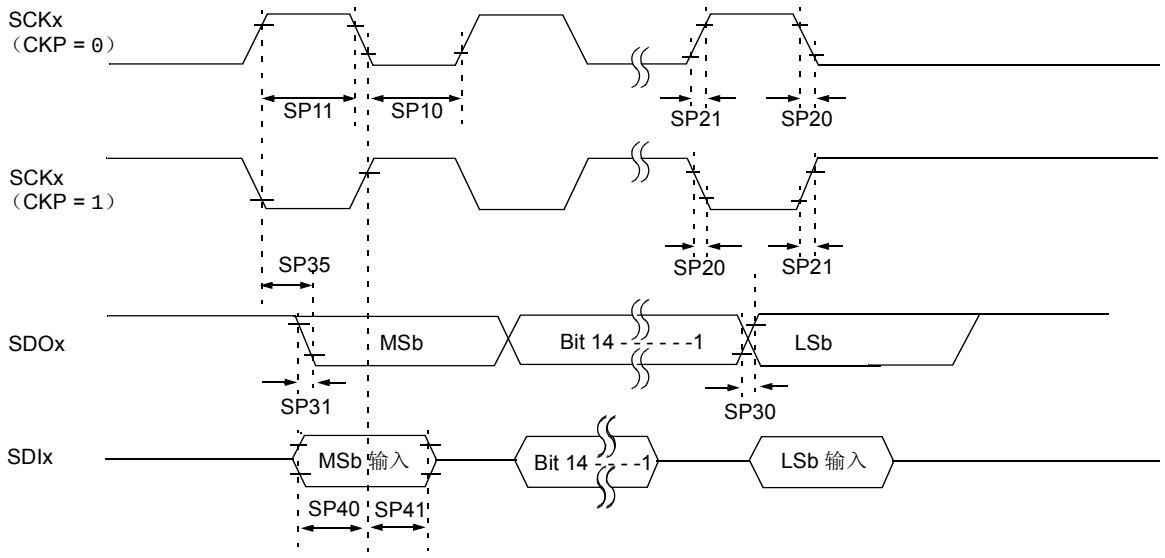
表 24-27：　　简单输出比较 /PWM 模式时序要求

交流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度　　-40°C ≤ TA ≤ +85°C				
参数编号	符号	特性 (1)	最小值	典型值	最大值	单位	条件
OC15	TFD	故障输入到 PWM I/O 发生变化的时间	—	—	50	ns	—
OC20	TFLT	故障输入脉冲宽度	50	—	—	ns	—

注　1： 这些参数为特性值，但生产时未经测试。

© 2007 Microchip Technology Inc.

图 24-9：　　SPIx 模块主模式 （CKE = 0）时序特性



注：负载条件请参见图 24-1。

表 24-28：　　SPIx 主模式 （CKE = 0）时序要求

交流特性			**标准工作条件：3.0V 至 3.6V（除非另外声明）** 工作温度　　-40°C ≤ TA ≤ +85°C				
参数编号	**符号**	**特性 (1)**	**最小值**	**典型值 (2)**	**最大值**	**单位**	**条件**
SP10	TscL	SCKx 输出低电平时间 (3)	TCY/2	—	—	ns	—
SP11	TscH	SCKx 输出高电平时间 (3)	TCY/2	—	—	ns	—
SP20	TscF	SCKx 输出下降时间 (4)	—	—	—	ns	见参数 D032
SP21	TscR	SCKx 输出上升时间 (4)	—	—	—	ns	见参数 D031
SP30	TdoF	SDOx 数据输出下降时间 (4)	—	—	—	ns	见参数 D032
SP31	TdoR	SDOx 数据输出上升时间 (4)	—	—	—	ns	见参数 D031
SP35	TscH2doV，TscL2doV	在 SCKx 边沿后 SDOx 数据输出有效的时间	—	6	20	ns	—
SP40	TdiV2scH，TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	23	—	—	ns	—
SP41	TscH2diL，TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	30	—	—	ns	—

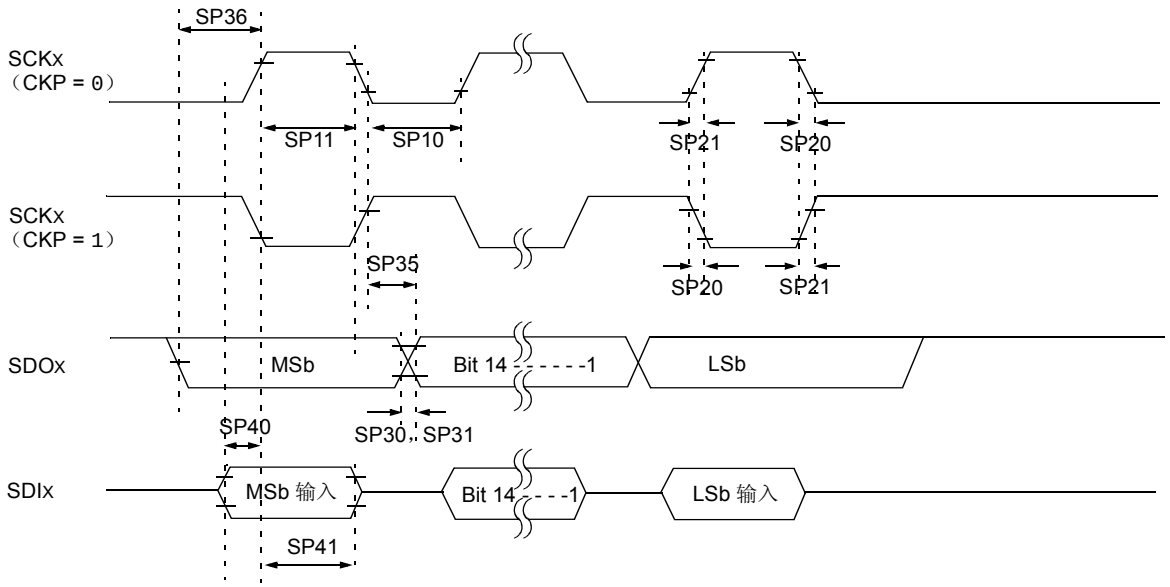
注　1： 这些参数为特性值，但生产时未经测试。

2： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

3： SCKx 的最小时钟周期为 100 ns。因此，主模式下产生的时钟不应违反此规范。

4： 假定所有 SPIx 引脚上的负载均为 50 pF。

图 24-10：　　SPIx 模块主模式 （CKE = 1）时序特性



注：负载条件请参见图 24-1。

表 24-29：　　SPIx 模块主模式 （CKE = 1）时序要求

交流特性			**标准工作条件：3.0V 至 3.6V（除非另外声明）** 工作温度　　-40°C ≤ TA ≤ +85°C				
参数编号	**符号**	**特性 (1)**	**最小值**	**典型值 (2)**	**最大值**	**单位**	**条件**
SP10	TscL	SCKx 输出低电平时间 (3)	TCY/2	—	—	ns	—
SP11	TscH	SCKx 输出高电平时间 (3)	TCY/2	—	—	ns	—
SP20	TscF	SCKx 输出下降时间 (4)	—	—	—	ns	见参数 D032
SP21	TscR	SCKx 输出上升时间 (4)	—	—	—	ns	见参数 D031
SP30	TdoF	SDOx 数据输出下降时间 (4)	—	—	—	ns	见参数 D032
SP31	TdoR	SDOx 数据输出上升时间 (4)	—	—	—	ns	见参数 D031
SP35	TscH2doV，TscL2doV	在 SCKx 边沿后 SDOx 数据输出有效的时间	—	6	20	ns	—
SP36	TdoV2sc，TdoV2scL	SDOx 数据输出建立到出现第一个 SCKx 边沿的时间	30	—	—	ns	—
SP40	TdiV2scH，TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	23	—	—	ns	—
SP41	TscH2diL，TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	30	—	—	ns	—

注　1： 这些参数为特性值，但生产时未经测试。

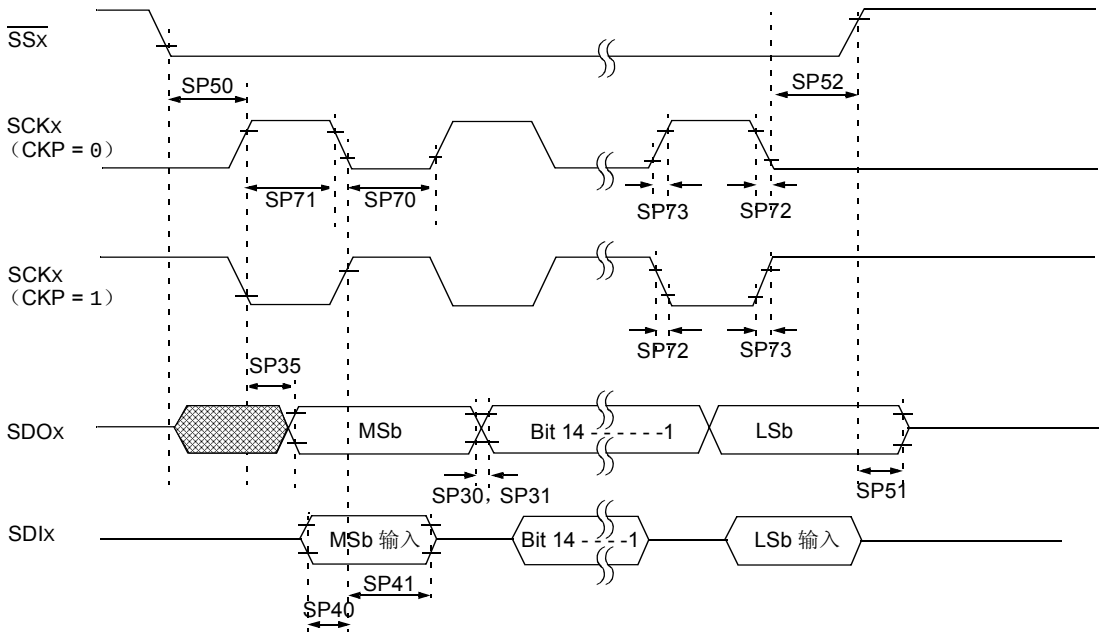
2： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

3： SCKx 的最小时钟周期为 100 ns。因此，主模式下产生的时钟不应违反此规范。

4： 假定所有 SPIx 引脚上的负载均为 50 pF。

© 2007 Microchip Technology Inc.

图 24-11： SPIx 模块从模式 （CKE = 0）时序特性



注：负载条件请参见图 24-1。

表 24-30： SPIx 模块从模式 （CKE = 0）时序要求

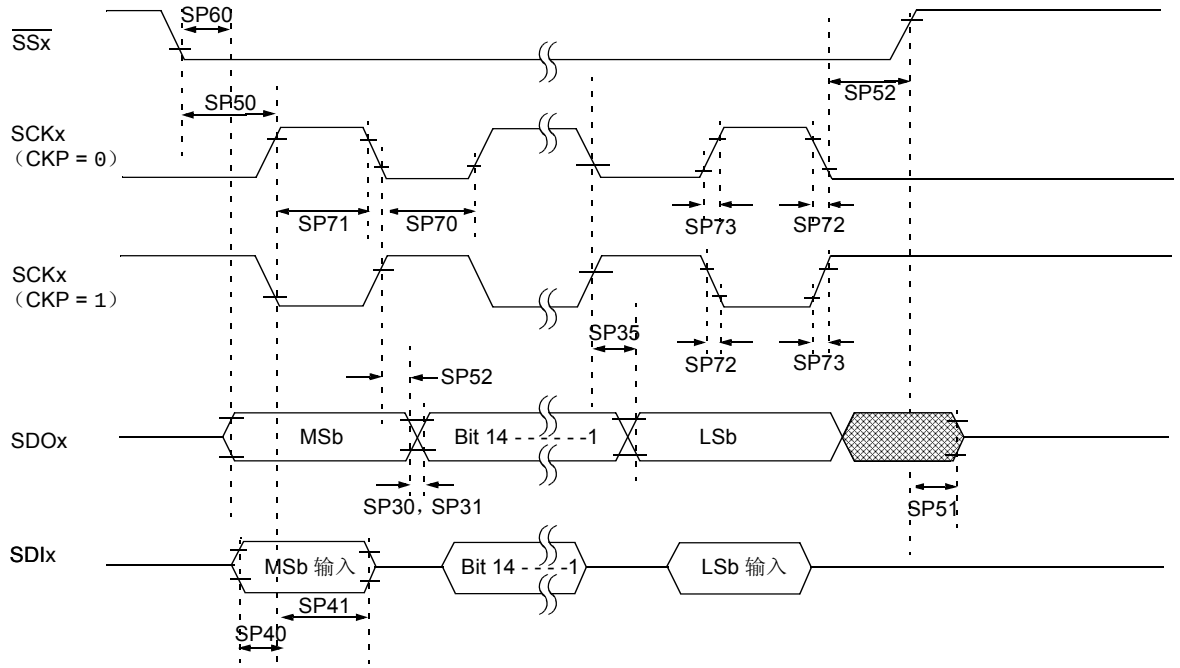
交流特性			**标准工作条件：3.0V 至 3.6V（除非另外声明）** 工作温度 -40°C ≤ TA ≤ +85°C				
参数编号	**符号**	**特性 (1)**	**最小值**	**典型值 (2)**	**最大值**	**单位**	**条件**
SP70	TscL	SCKx 输入低电平时间	30	—	—	ns	—
SP71	TscH	SCKx 输入高电平时间	30	—	—	ns	—
SP72	TscF	SCKx 输入下降时间 (3)	—	10	25	ns	—
SP73	TscR	SCKx 输入上升时间 (3)	—	10	25	ns	—
SP30	TdoF	SDOx 数据输出下降时间 (3)	—	—	—	ns	见参数 D032
SP31	TdoR	SDOx 数据输出上升时间 (3)	—	—	—	ns	见参数 D031
SP35	TscH2doV，TscL2doV	在 SCKx 边沿后 SDOx 数据输出有效的时间	—	—	30	ns	—
SP40	TdiV2scH，TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	20	—	—	ns	—
SP41	TscH2diL，TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	20	—	—	ns	—
SP50	TssL2scH，TssL2scL	$\overline{SSx}$ ↓ 到 SCKx ↑ 或 SCKx 输入的时间	120	—	—	ns	—
SP51	TssH2doZ	$\overline{SSx}$ ↑ 到 SDOx 输出高阻态的时间 (3)	10	—	50	ns	—
SP52	TscH2ssH TscL2ssH	SCKx 边沿后 $\overline{SSx}$ 有效的时间	1.5 TCY +40	—	—	ns	—

注 1： 这些参数为特性值，但生产时未经测试。

2： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

3： 假定所有 SPIx 引脚上的负载均为 50 pF。

图 24-12： SPIx 模块从模式 （CKE = 1）时序特性



注： 负载条件请参见图 24-1。

表 24-31： SPIx 模块从模式 （CKE = 1）时序要求

交流特性			**标准工作条件：3.0V 至 3.6V（除非另外声明）** 工作温度 -40°C ≤ TA ≤ +85°C				
参数编号	**符号**	**特性 (1)**	**最小值**	**典型值 (2)**	**最大值**	**单位**	**条件**
SP70	TscL	SCKx 输入低电平时间	30	—	—	ns	—
SP71	TscH	SCKx 输入高电平时间	30	—	—	ns	—
SP72	TscF	SCKx 输入下降时间 (3)	—	10	25	ns	—
SP73	TscR	SCKx 输入上升时间 (3)	—	10	25	ns	—
SP30	TdoF	SDOx 数据输出下降时间 (3)	—	—	—	ns	见参数 D032
SP31	TdoR	SDOx 数据输出上升时间 (3)	—	—	—	ns	见参数 D031
SP35	TscH2doV，TscL2doV	在 SCKx 边沿后 SDOx 数据输出有效的时间	—	—	30	ns	—
SP40	TdiV2scH，TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	20	—	—	ns	—
SP41	TscH2diL，TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	20	—	—	ns	—
SP50	TssL2scH，TssL2scL	$\overline{SSx}$↓到 SCKx↓或 SCKx↑输入的时间	120	—	—	ns	—
SP51	TssH2doZ	$\overline{SSx}$↑到 SDOx 输出高阻态的时间 (4)	10	—	50	ns	—

注 1： 这些参数为特性值，但生产时未经测试。

2： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

3： SCKx 的最小时钟周期为 100 ns。因此，主模式下产生的时钟不应违反此规范。

4： 假定所有 SPIx 引脚上的负载均为 50 pF。

© 2007 Microchip Technology Inc.

表 24-31： SPIx 模块从模式 （CKE = 1）时序要求 （续）

交流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP52	TscH2ssH TscL2ssH	SCKx 边沿后 $\overline{SSx}$ ↑ 有效的时间	1.5 TCY + 40	—	—	ns	—
SP60	TssL2doV	在 $\overline{SSx}$ 边沿后 SDOx 数据输出有效的时间	—	—	50	ns	—

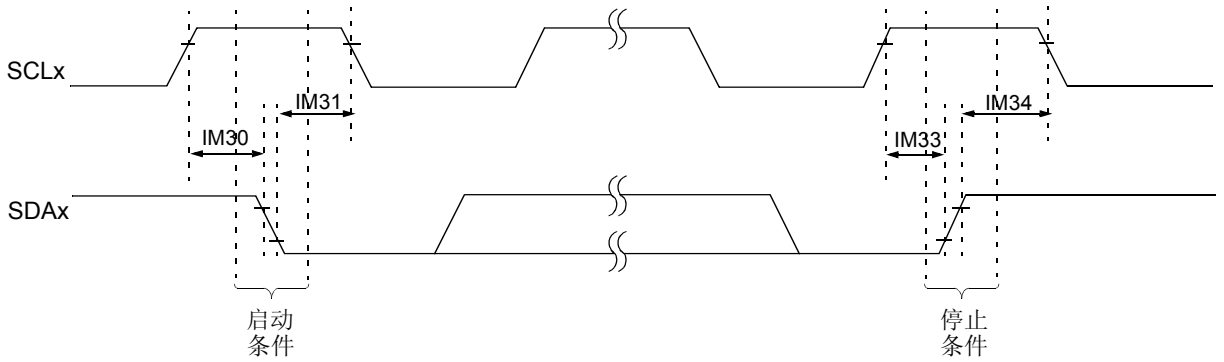
注 1： 这些参数为特性值，但生产时未经测试。

2： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

3： SCKx 的最小时钟周期为 100 ns。因此，主模式下产生的时钟不应违反此规范。

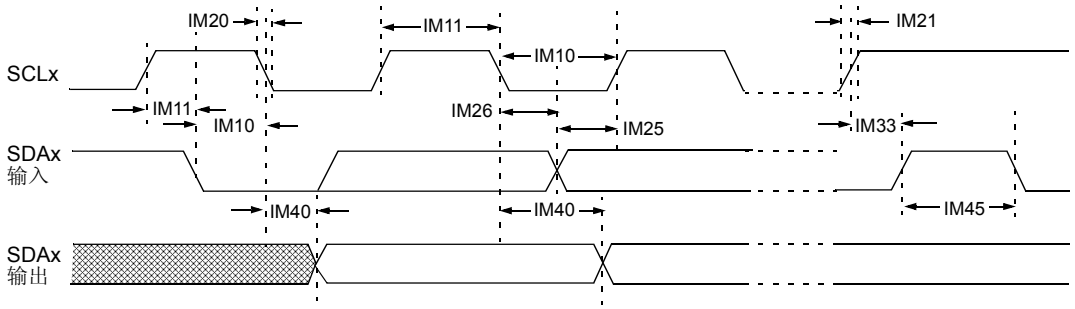
4： 假定所有 SPIx 引脚上的负载均为 50 pF。

图 24-13：　　I2Cx 总线启动位 / 停止位时序特性 （主模式）



注：负载条件请参见图 24-1。

图 24-14：　　I2Cx 总线数据时序特性 （主模式）



注：负载条件请参见图 24-1。

© 2007 Microchip Technology Inc.

表 24-32： I2Cx 总线数据时序要求 （主模式）

交流特性				标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C			
参数编号	符号	特性		最小值 (1)	最大值	单位	条件
IM10	TLO:SCL	时钟低电平时间	100 kHz 模式	TCY/2 (BRG + 1)	—	µs	—
			400 kHz 模式	TCY/2 (BRG + 1)	—	µs	—
			1 MHz 模式 (2)	TCY/2 (BRG + 1)	—	µs	—
IM11	THI:SCL	时钟高电平时间	100 kHz 模式	TCY/2 (BRG + 1)	—	µs	—
			400 kHz 模式	TCY/2 (BRG + 1)	—	µs	—
			1 MHz 模式 (2)	TCY/2 (BRG + 1)	—	µs	—
IM20	TF:SCL	SDAx 和 SCLx 下降时间	100 kHz 模式	—	300	ns	CB 值规定在 10 至 400pF 之间
			400 kHz 模式	20 + 0.1 CB	300	ns	
			1 MHz 模式 (2)	—	100	ns	
IM21	TR:SCL	SDAx 和 SCLx 上升时间	100 kHz 模式	—	1000	ns	CB 值规定在 10 至 400pF 之间
			400 kHz 模式	20 + 0.1 CB	300	ns	
			1 MHz 模式 (2)	—	300	ns	
IM25	TSU:DAT	数据输入建立时间	100 kHz 模式	250	—	ns	—
			400 kHz 模式	100	—	ns	
			1 MHz 模式 (2)	40	—	ns	
IM26	THD:DAT	数据输入保持时间	100 kHz 模式	0	—	µs	—
			400 kHz 模式	0	0.9	µs	
			1 MHz 模式 (2)	0.2	—	µs	
IM30	TSU:STA	启动条件建立时间	100 kHz 模式	TCY/2 (BRG + 1)	—	µs	仅与重复启动条件相关
			400 kHz 模式	TCY/2 (BRG + 1)	—	µs	
			1 MHz 模式 (2)	TCY/2 (BRG + 1)	—	µs	
IM31	THD:STA	启动条件保持时间	100 kHz 模式	TCY/2 (BRG + 1)	—	µs	这个周期后产生第一个时钟脉冲
			400 kHz 模式	TCY/2 (BRG + 1)	—	µs	
			1 MHz 模式 (2)	TCY/2 (BRG + 1)	—	µs	
IM33	TSU:STO	停止条件建立时间	100 kHz 模式	TCY/2 (BRG + 1)	—	µs	—
			400 kHz 模式	TCY/2 (BRG + 1)	—	µs	
			1 MHz 模式 (2)	TCY/2 (BRG + 1)	—	µs	
IM34	THD:STO	停止条件保持时间	100 kHz 模式	TCY/2 (BRG + 1)	—	ns	—
			400 kHz 模式	TCY/2 (BRG + 1)	—	ns	
			1 MHz 模式 (2)	TCY/2 (BRG + 1)	—	ns	
IM40	TAA:SCL	自时钟边沿到输出有效的时间	100 kHz 模式	—	3500	ns	—
			400 kHz 模式	—	1000	ns	—
			1 MHz 模式 (2)	—	400	ns	—
IM45	TBF:SDA	总线空闲时间	100 kHz 模式	4.7	—	µs	在启动一个新的传输前总线必须保持空闲的时间
			400 kHz 模式	1.3	—	µs	
			1 MHz 模式 (2)	0.5	—	µs	
IM50	CB	总线容性负载		—	400	pF	—

注 1： BRG 为 I^2C 波特率发生器的值。请参见 《dsPIC33F 系列参考手册》中的**第 19 章 “I^2C™”**。

2： 所有 I2Cx 引脚的最大引脚电容为 10 pF （仅对于 1 MHz 模式）。

图 24-15： I2Cx 总线启动位 / 停止位时序特性 （从模式）

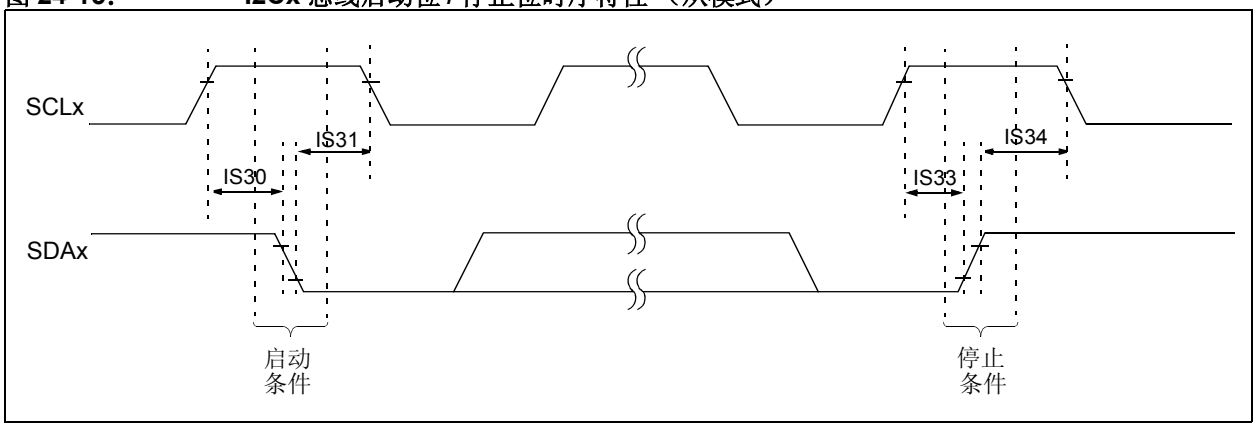
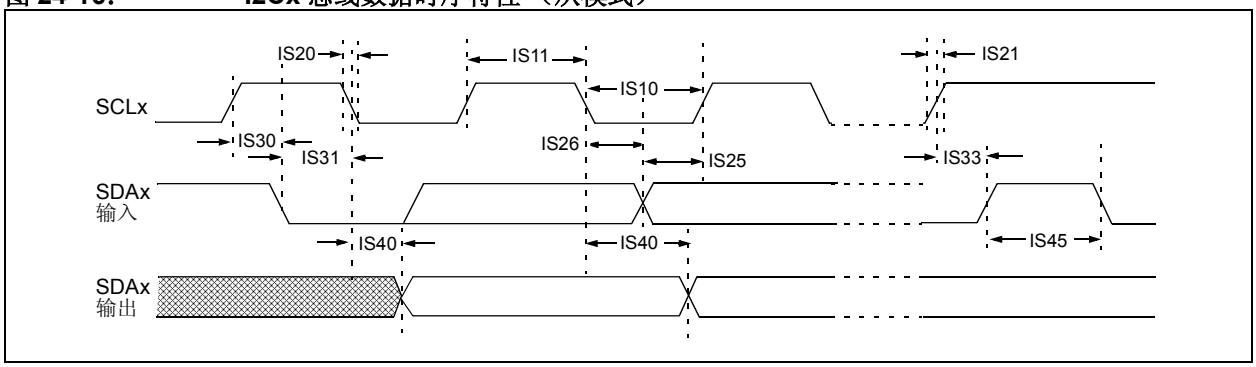


图 24-16： I2Cx 总线数据时序特性 （从模式）



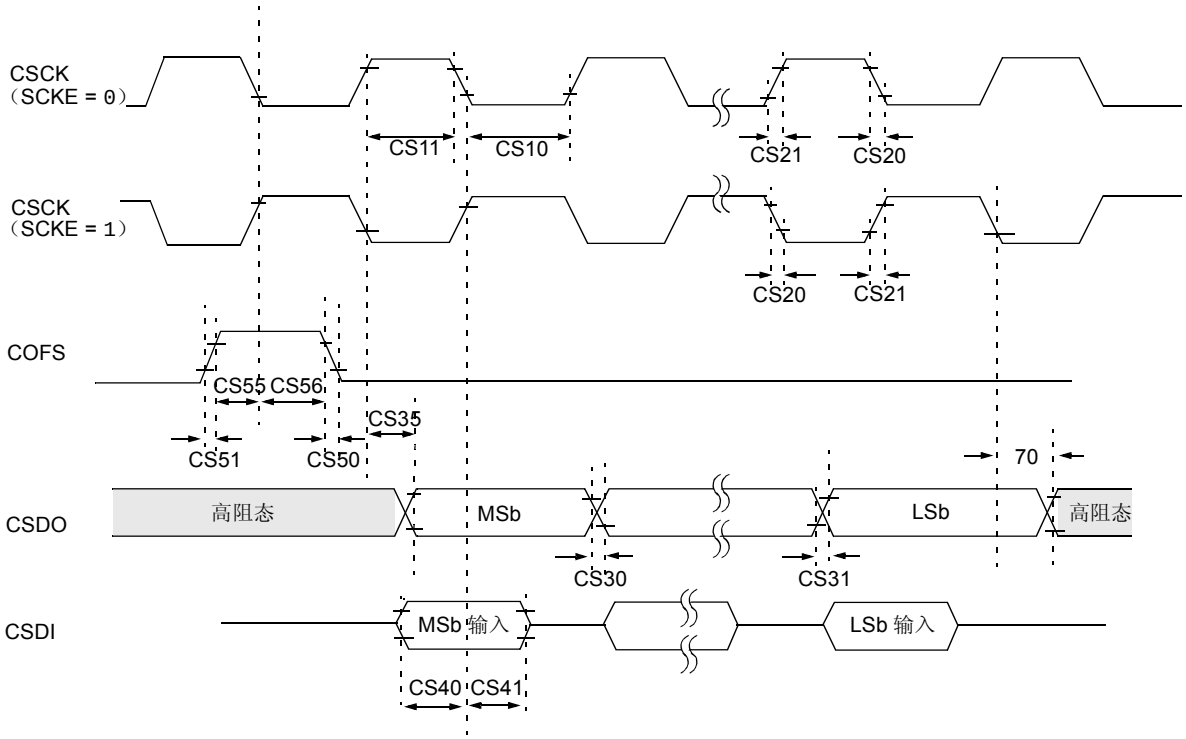
© 2007 Microchip Technology Inc.

表 24-33： I2Cx 总线数据时序要求 （从模式）

交流特性				标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C			
参数编号	符号	特性		最小值	最大值	单位	条件
IS10	TLO:SCL	时钟低电平时间	100 kHz 模式	4.7	—	µs	器件工作频率不得低于 1.5 MHz
			400 kHz 模式	1.3	—	µs	器件工作频率不得低于 10 MHz
			1 MHz 模式 (1)	0.5	—	µs	—
IS11	THI:SCL	时钟高电平时间	100 kHz 模式	4.0	—	µs	器件工作频率不得低于 1.5 MHz
			400 kHz 模式	0.6	—	µs	器件工作频率不得低于 10 MHz
			1 MHz 模式 (1)	0.5	—	µs	—
IS20	TF:SCL	SDAx 和 SCLx 下降时间	100 kHz 模式	—	300	ns	CB 值规定在 10 至 400 pF 之间
			400 kHz 模式	20 + 0.1 CB	300	ns	
			1 MHz 模式 (1)	—	100	ns	
IS21	TR:SCL	SDAx 和 SCLx 上升时间	100 kHz 模式	—	1000	ns	CB 值规定在 10 至 400 pF 之间
			400 kHz 模式	20 + 0.1 CB	300	ns	
			1 MHz 模式 (1)	—	300	ns	
IS25	TSU:DAT	数据输入建立时间	100 kHz 模式	250	—	ns	—
			400 kHz 模式	100	—	ns	
			1 MHz 模式 (1)	100	—	ns	
IS26	THD:DAT	数据输入保持时间	100 kHz 模式	0	—	µs	—
			400 kHz 模式	0	0.9	µs	
			1 MHz 模式 (1)	0	0.3	µs	
IS30	TSU:STA	启动条件建立时间	100 kHz 模式	4.7	—	µs	仅与重复启动条件相关
			400 kHz 模式	0.6	—	µs	
			1 MHz 模式 (1)	0.25	—	µs	
IS31	THD:STA	启动条件保持时间	100 kHz 模式	4.0	—	µs	这个周期后产生第一个时钟脉冲
			400 kHz 模式	0.6	—	µs	
			1 MHz 模式 (1)	0.25	—	µs	
IS33	TSU:STO	停止条件建立时间	100 kHz 模式	4.7	—	µs	—
			400 kHz 模式	0.6	—	µs	
			1 MHz 模式 (1)	0.6	—	µs	
IS34	THD:STO	停止条件保持时间	100 kHz 模式	4000	—	ns	—
			400 kHz 模式	600	—	ns	
			1 MHz 模式 (1)	250	—	ns	
IS40	TAA:SCL	自时钟边沿到输出有效的时间	100 kHz 模式	0	3500	ns	—
			400 kHz 模式	0	1000	ns	
			1 MHz 模式 (1)	0	350	ns	
IS45	TBF:SDA	总线空闲时间	100 kHz 模式	4.7	—	µs	在启动一个新的传输前总线必须保持空闲的时间
			400 kHz 模式	1.3	—	µs	
			1 MHz 模式 (1)	0.5	—	µs	
IS50	CB	总线容性负载		—	400	pF	—

注 1： 所有 I2Cx 引脚的最大引脚电容为 10 pF （仅对于 1 MHz 模式）。

图 24-17： DCI 模块 （多通道， I²S 模式）时序特性



注： 负载条件请参见图 24-1。

© 2007 Microchip Technology Inc.

表 24-34： DCI 模块 （多通道， I²S 模式）时序要求

交流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
CS10	TCSCKL	CSCK 输入低电平时间（CSCK 引脚为输入）	TCY/2 + 20	—	—	ns	—
		CSCK 输出低电平时间 (3)（CSCK 引脚为输出）	30	—	—	ns	—
CS11	TCSCKH	CSCK 输入高电平时间（CSCK 引脚为输入）	TCY/2 + 20	—	—	ns	—
		CSCK 输出高电平时间 (3)（CSCK 引脚为输出）	30	—	—	ns	—
CS20	TCSCKF	CSCK 输出下降时间 (4)（CSCK 引脚为输出）	—	10	25	ns	—
CS21	TCSCKR	CSCK 输出上升时间 (4)（CSCK 引脚为输出）	—	10	25	ns	—
CS30	TCSDOF	CSDO 数据输出下降时间 (4)	—	10	25	ns	—
CS31	TCSDOR	CSDO 数据输出上升时间 (4)	—	10	25	ns	—
CS35	TDV	时钟边沿到 CSDO 数据有效的时间	—	—	10	ns	—
CS36	TDIV	时钟边沿到 CSDO 三态的时间	10	—	20	ns	—
CS40	TCSDI	CSDI 数据输入到 CSCK 边沿的建立时间 （CSCK 引脚为输入或输出）	20	—	—	ns	—
CS41	THCSDI	CSDI 数据输入到 CSCK 边沿的保持时间 （CSCK 引脚为输入或输出）	20	—	—	ns	—
CS50	TCOFSF	COFS 下降时间（COFS 引脚为输出）	—	10	25	ns	**注 1**
CS51	TCOFSR	COFS 上升时间（COFS 引脚为输出）	—	10	25	ns	**注 1**
CS55	TSCOFS	COFS 数据输入到 CSCK 边沿的建立时间 （COFS 引脚为输入）	20	—	—	ns	—
CS56	THCOFS	COFS 数据输入到 CSCK 边沿的保持时间 （COFS 引脚为输入）	20	—	—	ns	—

注 1： 这些参数为特性值，但生产时未经测试。

2： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。这些参数仅供设计参考，未经测试。

3： CSCK 的最小时钟周期为 100 ns。因此，主模式下产生的时钟不应违反此规范。

4： 假定所有 DCI 引脚上的负载均为 50 pF。

图 24-18：　　DCI 模块 （AC-LINK 模式）时序特性

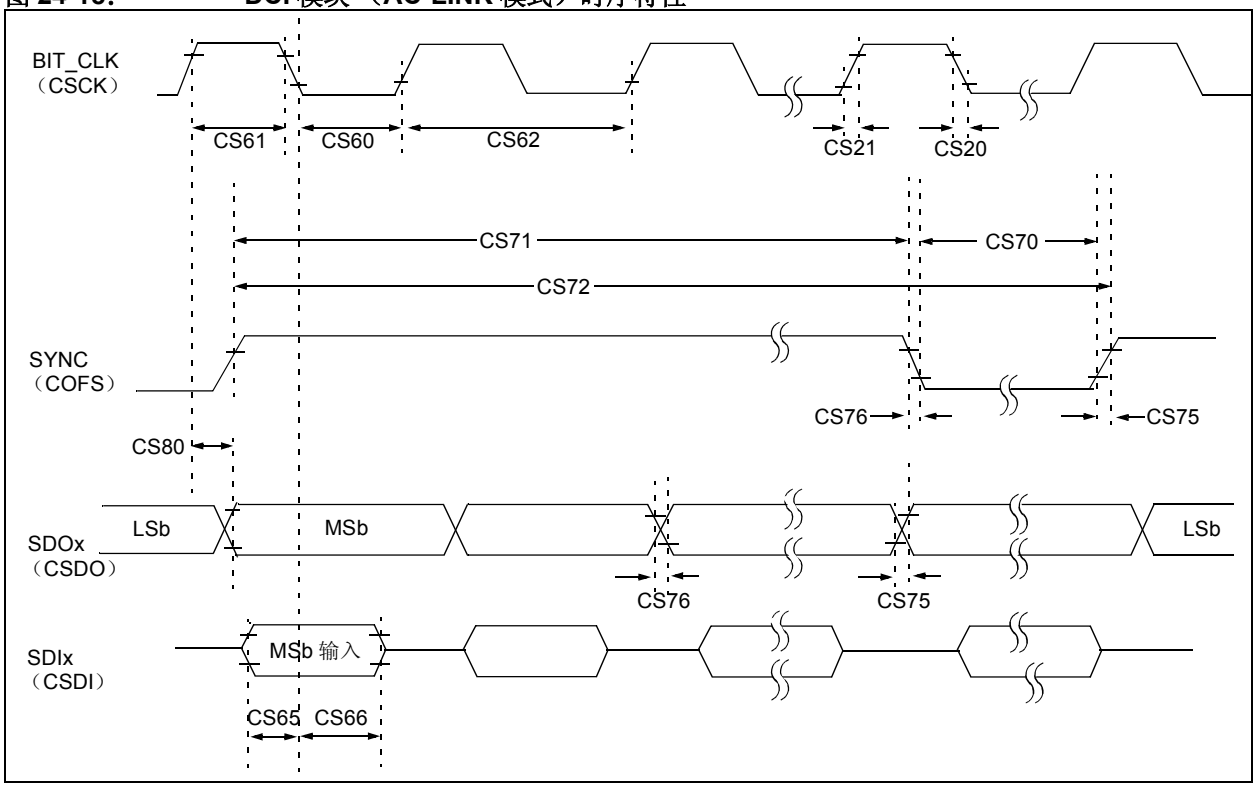


表 24-35：　　DCI 模块 （AC-LINK 模式）时序要求

交流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度　　-40°C ≤ TA ≤ +85°C				
参数编号	符号	特性 (1,2)	最小值	典型值 (3)	最大值	单位	条件
CS60	TBCLKL	BIT_CLK 低电平时间	36	40.7	45	ns	—
CS61	TBCLKH	BIT_CLK 高电平时间	36	40.7	45	ns	—
CS62	TBCLK	BIT_CLK 周期	—	81.4	—	ns	位时钟为输入
CS65	TSACL	到 BIT_CLK 下降沿的输入建立时间	—	—	10	ns	—
CS66	THACL	自 BIT_CLK 下降沿起的输入保持时间	—	—	10	ns	—
CS70	TSYNCLO	同步数据输出低电平时间	—	19.5	—	µs	**注 1**
CS71	TSYNCHI	同步数据输出高电平时间	—	1.3	—	µs	**注 1**
CS72	TSYNC	同步数据输出周期	—	20.8	—	µs	**注 1**
CS75	TRACL	上升时间，同步， SDATA_OUT	—	10	25	ns	CLOAD = 50 pF，VDD = 5V
CS76	TFACL	下降时间，同步， SDATA_OUT	—	10	25	ns	CLOAD = 50 pF，VDD = 5V
CS77	TRACL	上升时间，同步， SDATA_OUT	—	—	30	ns	CLOAD = 50 pF，VDD = 3V
CS78	TFACL	下降时间，同步， SDATA_OUT	—	—	30	ns	CLOAD = 50 pF，VDD = 3V
CS80	TOVDACL	自 BIT_CLK 上升沿起的输出有效延时	—	—	15	ns	—

注　1： 这些参数为特性值，但生产时未经测试。

2： 这些值假设 BIT_CLK 频率为 12.288 MHz。

3： 除非另外声明，否则 “典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。这些参数仅供设计参考，未经测试。

© 2007 Microchip Technology Inc.

图 24-19：　　CAN 模块 I/O 时序特性

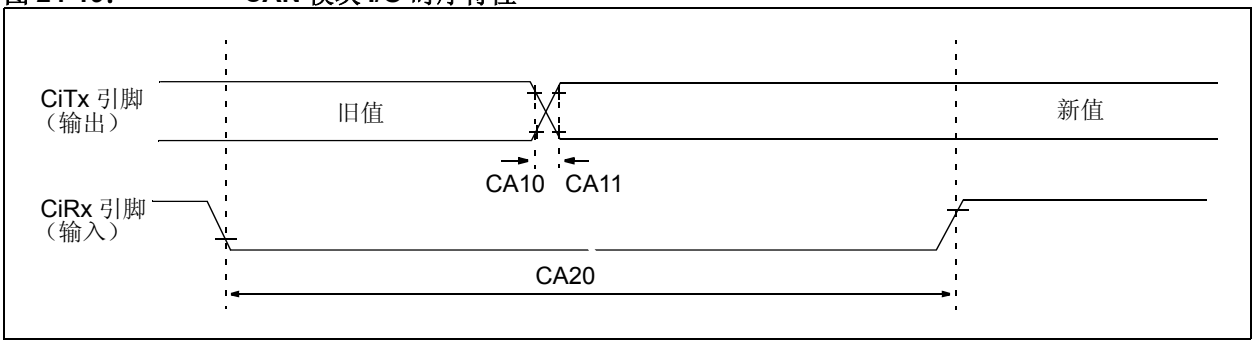


表 24-36：　　CAN 模块 I/O 时序要求

交流特性			**标准工作条件：3.0V 至 3.6V（除非另外声明）** 工作温度　　-40°C ≤ TA ≤ +85°C				
参数编号	**符号**	**特性 (1)**	**最小值**	**典型值**	**最大值**	**单位**	**条件**
CA10	TioF	端口输出下降时间	—	—	—	ns	见参数 D032
CA11	TioR	端口输出上升时间	—	—	—	ns	见参数 D031
CA20	Tcwf	触发 CAN 唤醒滤波器的脉冲宽度	120	—	—	ns	—

注　1： 这些参数为特性值，但生产时未经测试。

表 24-37： ADC 模块规范

交流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级） -40°C ≤ TA ≤ +125°C （扩展级）				
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
器件电源							
AD01	AVDD	模块电源 VDD	VDD – 0.3 和 3.0 中的较大值	—	VDD + 0.3 和 3.6 中的较小值	V	—
AD02	AVSS	模块电源 VSS	VSS – 0.3	—	VSS + 0.3	V	—
参考输入							
AD05	VREFH	参考电压高电平	AVSS + 2.7	—	AVDD	V	见**注 2**
AD05a			3.0	—	3.6	V	VREFH = AVDD VREFL = AVSS = 0
AD06	VREFL	参考电压低电平	AVSS	—	AVDD – 2.7	V	见**注 2**
AD06a			0	—	0	V	VREFH = AVDD VREFL = AVSS = 0
AD07	VREF	绝对参考电压	3.0	—	3.6	V	VREF = VREFH - VREFL
AD08	IREF	电流消耗	—	389 .001	549 1	μA μA	ADC 工作 ADC 关闭
模拟输入							
AD12	VINH	输入电压范围 VINH	VINL	—	VREFH	V	该电压反映了采样和保持通道 0、 1、 2 和 3（CH0-CH3），正输入。见**注 1**
AD13	VINL	输入电压范围 VINL	VREFL	—	AVSS + 1V	V	该电压反映了采样和保持通道 0、 1、 2 和 3（CH0-CH3），负输入。见**注 1**
AD17	RIN	模拟输入信号源的推荐阻抗	—	—	200 200	Ω Ω	10 位 12 位

注 1： ADC 转换结果不会因输入电压的增加而减小，并且不会丢失编码。

2： 这些参数不是特性值，或生产时未经测试。

© 2007 Microchip Technology Inc.

表 24-38： ADC 模块规范 （12 位模式）

交流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级） -40°C ≤ TA ≤ +125°C （扩展级）				
参数编号	**符号**	**特性**	**最小值**	**典型值**	**最大值**	**单位**	**条件**
ADC 精度 （12 位模式）——测量采用外部 VREF+/VREF-							
AD20a	Nr	分辨率	12 个数据位			位	
AD21a	INL	积分非线性误差	-1	—	+1	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD22a	DNL	微分非线性误差	>-1	—	<1	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD23a	GERR	增益误差	1.25	1.5	3	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD24a	EOFF	失调误差	-2	-1.5	-1.25	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD25a		单调性 (1)	—	—	—	—	保证
ADC 精度 （12 位模式）——测量采用内部 VREF+/VREF-							
AD20a	Nr	分辨率	12 个数据位			位	
AD21a	INL	积分非线性误差	-1	—	+1	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD22a	DNL	微分非线性误差	>-1	—	<1	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD23a	GERR	增益误差	2	3	7	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD24a	EOFF	失调误差	2	3	5	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD25a		单调性 (1)	—	—	—	—	保证
动态性能 （12 位模式）							
AD30a	THD	总谐波失真	-77	-69	-61	dB	—
AD31a	SINAD	信噪比和失真	59	63	64	dB	—
AD32a	SFDR	无杂散动态范围	63	72	79	dB	—
AD33a	FNYQ	输入信号带宽	—	—	250	kHz	—
AD34a	ENOB	有效位数	10.95	11.1	—	位	—

注 1： ADC 转换结果不会因输入电压的增加而减小，并且不会丢失编码。

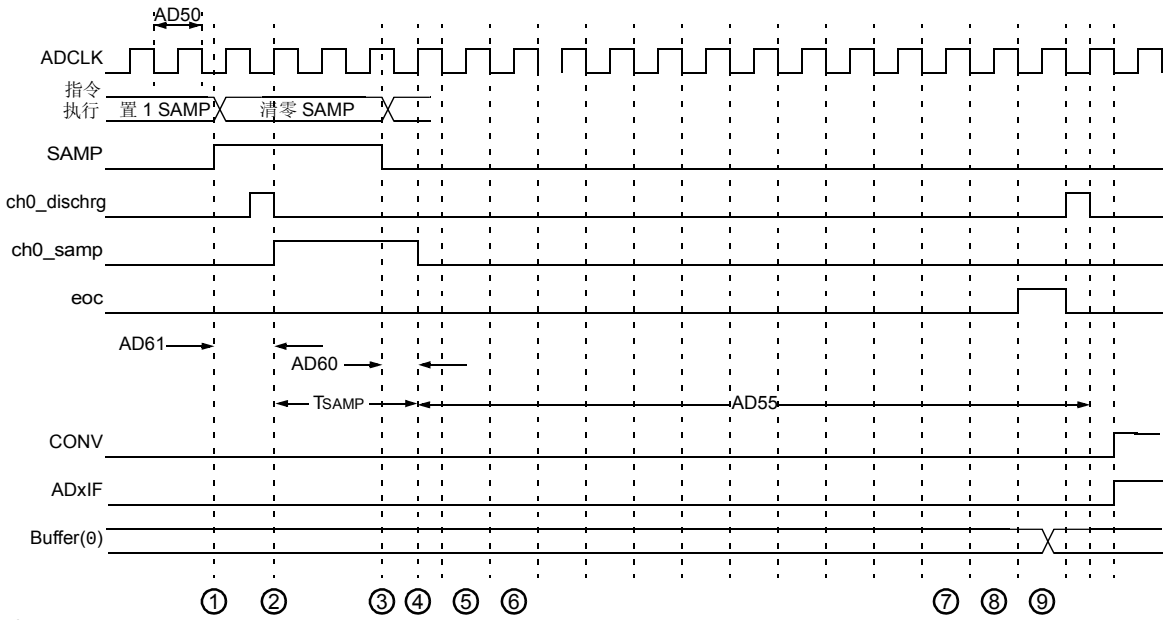
表 24-39： ADC 模块规范 （10 位模式）

交流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级） -40°C ≤ TA ≤ +125°C （扩展级）				
参数编号	**符号**	**特性**	**最小值**	**典型值**	**最大值**	**单位**	**条件**
ADC 精度 （10 位模式）——测量采用外部 VREF+/VREF-							
AD20b	Nr	分辨率	10 个数据位			位	
AD21b	INL	积分非线性误差	-1	—	+1	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD22b	DNL	微分非线性误差	>-1	—	<1	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD23b	GERR	增益误差	1	3	6	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD24b	EOFF	失调误差	1	2	5	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD25b		单调性 (1)	—	—	—	—	保证
ADC 精度 （10 位模式）——测量采用内部 VREF+/VREF-							
AD20b	Nr	分辨率	10 个数据位			位	
AD21b	INL	积分非线性误差	-1	—	+1	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD22b	DNL	微分非线性误差	>-1	—	<1	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD23b	GERR	增益误差	1	5	6	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD24b	EOFF	失调误差	1	2	3	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3.6V
AD25b		单调性 (1)	—	—	—	—	保证
动态性能 （10 位模式）							
AD30b	THD	总谐波失真	—	-64	-67	dB	—
AD31b	SINAD	信噪比和失真	—	57	58	dB	—
AD32b	SFDR	无杂散动态范围	—	67	71	dB	—
AD33b	FNYQ	输入信号带宽	—	—	550	kHz	—
AD34b	ENOB	有效位数	9.1	9.7	9.8	位	—

注 1： ADC 转换结果不会因输入电压的增加而减小，并且不会丢失编码。

© 2007 Microchip Technology Inc.

图 24-20： ADC 转换 （12 位模式）时序特性 （ASAM = 0， SSRC<2:0> = 000）



① – 软件置 1 ADxCON.SAMP 以启动采样。

② – 采样在放电周期之后启动。TSAMP 在《DSPIC33F 系列参考手册》的**第 16 章 “具有 DMA 的 10/12 位 ADC”**中进行介绍。

③ – 软件清零 ADxCON.SAMP 以启动转换。

④ – 采样结束，转换过程启动。

⑤ – 转换 bit 11。

⑥ – 转换 bit 10。

⑦ – 转换 bit 1。

⑧ – 转换 bit 0。

⑨ – 转换结束的一个 TAD。

表 24-40： ADC 转换 （12 位模式）时序要求

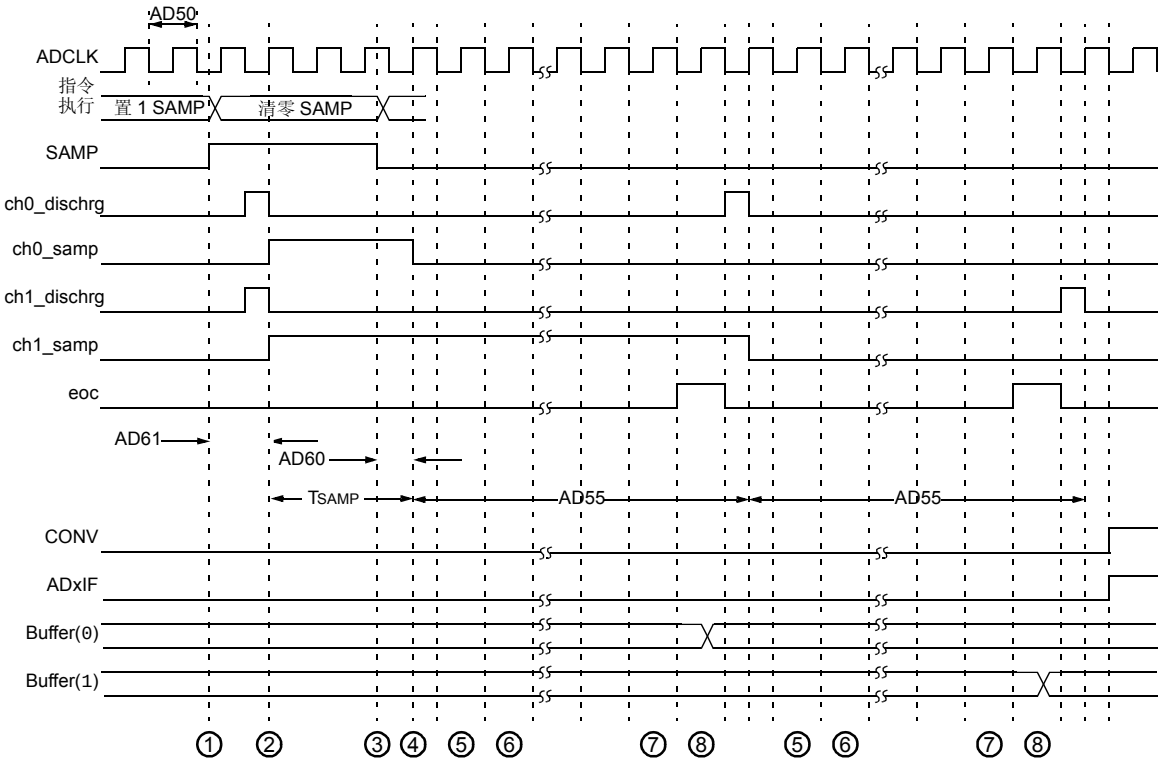
交流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C				
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
时钟参数							
AD50a	TAD	ADC 时钟周期	117.6	—	—	ns	
AD51a	tRC	ADC 内部 RC 振荡器周期	—	250	—	ns	
转换速率							
AD55a	tCONV	转换时间	—	14 TAD	—	ns	
AD56a	FCNV	吞吐率	—	—	500	KSPS	
AD57a	TSAMP	采样时间	3 TAD	—	—	—	
时序参数							
AD60a	tPCS	从采样触发到转换开始的时间 (1)	—	1.0 TAD	—	—	未选择自动转换触发（SSRC<2:0> = 111）
AD61a	tPSS	从采样位 （SAMP）置 1 到采样启动的时间 (1)	0.5 TAD	—	1.5 TAD	—	—
AD62a	tCSS	转换结束到采样启动 （ASAM = 1）的时间 (1)	—	0.5 TAD	—	—	—
AD63a	tDPU	从 ADC 关闭到 ADC 开启，用于稳定模拟级的时间 (1)	1	—	5	μs	—

注 1： 这些参数为特性值，但生产时未经测试。

2： 因为采样电容最终将释放电荷，因此低于 10 kHz 的时钟频率可能影响线性性能，尤其是在温度较高时。

© 2007 Microchip Technology Inc.

图 24-21：　　ADC 转换 （10 位模式）时序特性
（CHPS<1:0> = 01， SIMSAM = 0， ASAM = 0， SSRC<2:0> = 000）



① – 软件置 1 ADxCON.SAMP 以启动采样。

② – 采样在放电周期之后启动。TSAMP 在《dsPIC33F 系列参考手册》的 **第 16 章“具有 DMA 的 10/12 位 ADC”**中进行介绍。

③ – 软件清零 ADxCON.SAMP 以启动转换。

④ – 采样结束，转换过程启动。

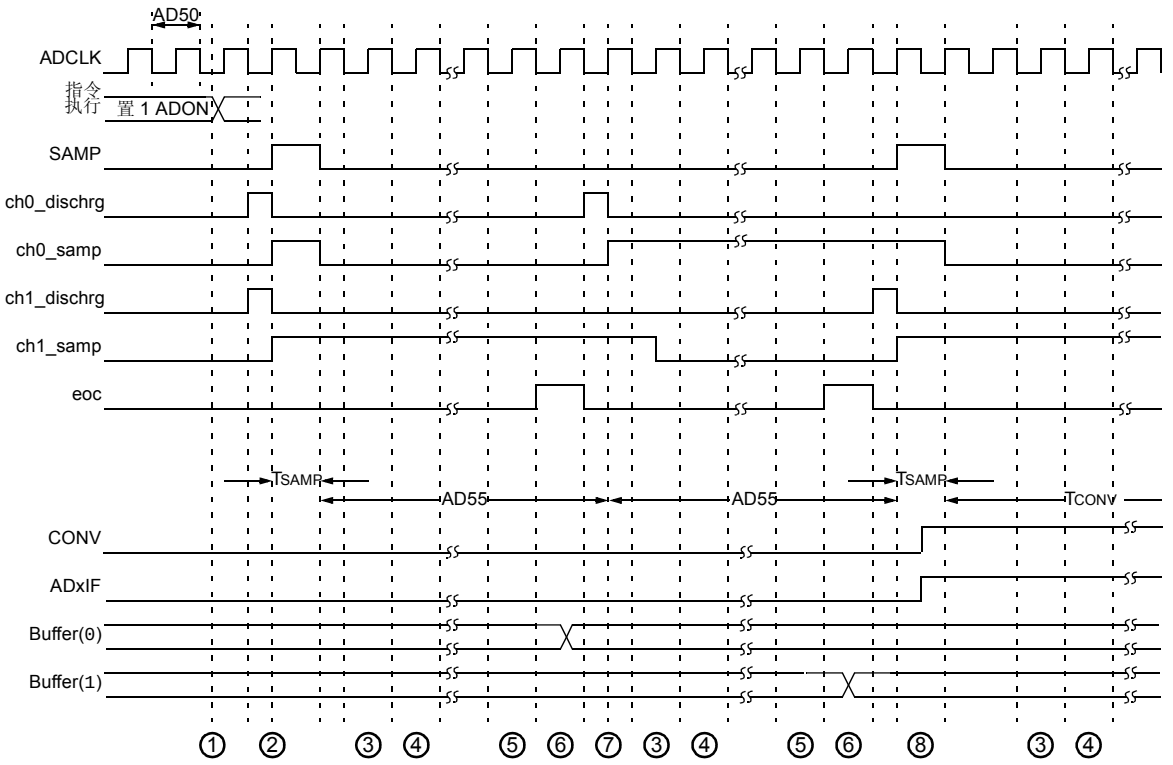
⑤ – 转换 bit 9。

⑥ – 转换 bit 8。

⑦ – 转换 bit 0。

⑧ – 转换结束的一个 TAD。

图 24-22：　ADC 转换 （10 位模式）时序特性 （CHPS<1:0> = 01， SIMSAM = 0， ASAM = 1， SSRC<2:0> = 111， SAMC<4:0> = 00001）



① – 软件置 1 ADxCON.ADON，使 AD 开始工作。

② – 采样在放电周期之后启动。TSAMP 在《dsPIC33F 系列参考手册》的**第 16 章 “具有 DMA 的 10/12 位 ADC”**中进行介绍。

③ – 转换 bit 9。

④ – 转换 bit 8。

⑤ – 转换 bit 0。

⑥ – 转换结束的一个 TAD。

⑦ – 开始下一个通道的转换。

⑧ – 采样时间由 SAMC<4:0> 指定。

© 2007 Microchip Technology Inc.

表 24-41： ADC 转换 （10 位模式）时序要求

交流特性			标准工作条件：3.0V 至 3.6V （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C				
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
时钟参数							
AD50b	TAD	ADC 时钟周期	65	—	—	ns	
AD51b	tRC	ADC 内部 RC 振荡器周期	—	250	—	ns	
转换速率							
AD55b	tCONV	转换时间	—	12 TAD	—	—	
AD56b	FCNV	吞吐率	—	—	1.1	MSPS	
AD57b	TSAMP	采样时间	2 TAD	—	—	—	
时序参数							
AD60b	tPCS	从采样触发到转换开始的时间 (3)	—	1.0 TAD	—	—	未选择自动转换触发（SSRC<2:0> = 111）
AD61b	tPSS	从采样位 （SAMP）置 1 到采样启动的时间 (1)	0.5 TAD	—	1.5 TAD	—	—
AD62b	tCSS	转换结束到采样启动 （ASAM = 1）的时间 (1)	—	0.5 TAD	—	—	—
AD63b	tDPU	从 ADC 关闭到 ADC 开启，用于稳定模拟级的时间 (1)	1	—	5	μs	—

注 1： 这些参数为特性值，但生产时未经测试。

2： 因为采样电容最终将释放电荷，因此低于 10 kHz 的时钟速率可能影响线性性能，尤其在温度较高时。

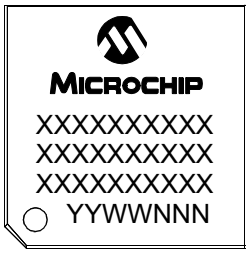
注：

© 2007 Microchip Technology Inc.

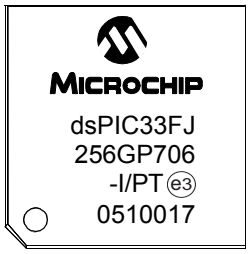
25.0 封装信息

25.1 封装标识信息

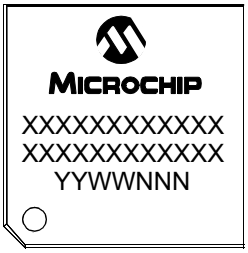
64 引脚 TQFP（10x10x1 mm）



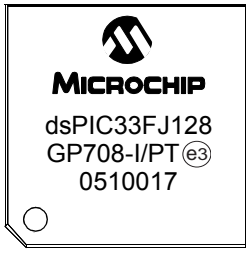
示例



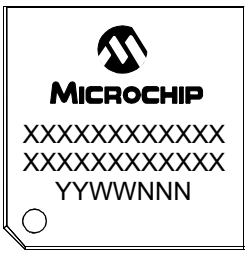
80 引脚 TQFP（12x12x1 mm）



示例



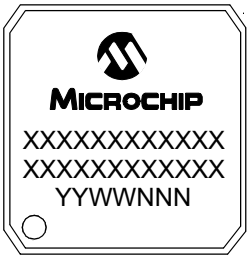
100 引脚 TQFP（12x12x1 mm）



示例



100 引脚 TQFP（14x14x1mm）



100 引脚 TQFP（14x14x1mm）



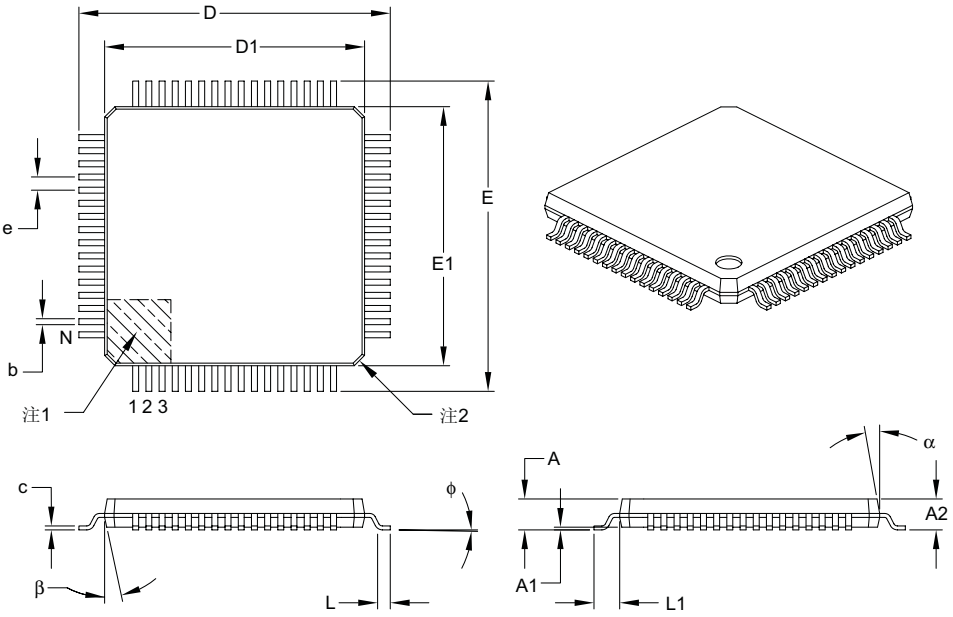
图注：	XX...X	客户信息
	Y	年份代码 （日历年的最后一位数字）
	YY	年份代码 （日历年的最后两位数字）
	WW	星期代码 （1 月 1 日的星期代码为 “01”）
	NNN	以字母数字排序的追踪代码
	e3	雾锡 （Matte Tin， Sn）的 JEDEC 无铅标志
	*	本封装为无铅封装。 JEDEC 无铅标志 （e3）标示于此种封装的外包装上。

注： Microchip 元器件编号如果无法在同一行内完整标注，将换行标出，因此会限制表示客户信息的字符数。

25.2 封装详细信息

64引脚塑封薄型正方扁平封装（**PT**）——**10x10x1 mm**主体，**2.00 mm**引脚投影长度 **[TQFP]**

注： 最新封装图请至http://www.microchip.com/packaging查看Microchip封装规范。



单位		毫米		
尺寸范围		最小	正常	最大
引脚数	N	64		
引脚间距	e	0.50 BSC		
总高度	A	–	–	1.20
塑模封装厚度	A2	0.95	1.00	1.05
悬空间隙	A1	0.05	–	0.15
底脚长度	L	0.45	0.60	0.75
引脚投影长度	L1	1.00 REF		
底脚倾斜角	φ	0°	3.5°	7°
总宽度	E	12.00 BSC		
总长度	D	12.00 BSC		
塑模封装宽度	E1	10.00 BSC		
塑模封装长度	D1	10.00 BSC		
引脚厚度	c	0.09	–	0.20
引脚宽度	b	0.17	0.22	0.27
塑模顶部锥度	α	11°	12°	13°
塑模底部锥度	β	11°	12°	13°

注：

1. 引脚1的可见定位功能可能不同，但必须在阴影区域内。
2. 切角处的斜面是可选的；大小可能有所不同。
3. 尺寸D1和E1不包括塑模毛边或突起。 塑模每侧的毛边或突起不得超过0.25毫米。
4. 尺寸和公差请参见ASME Y14.5M。

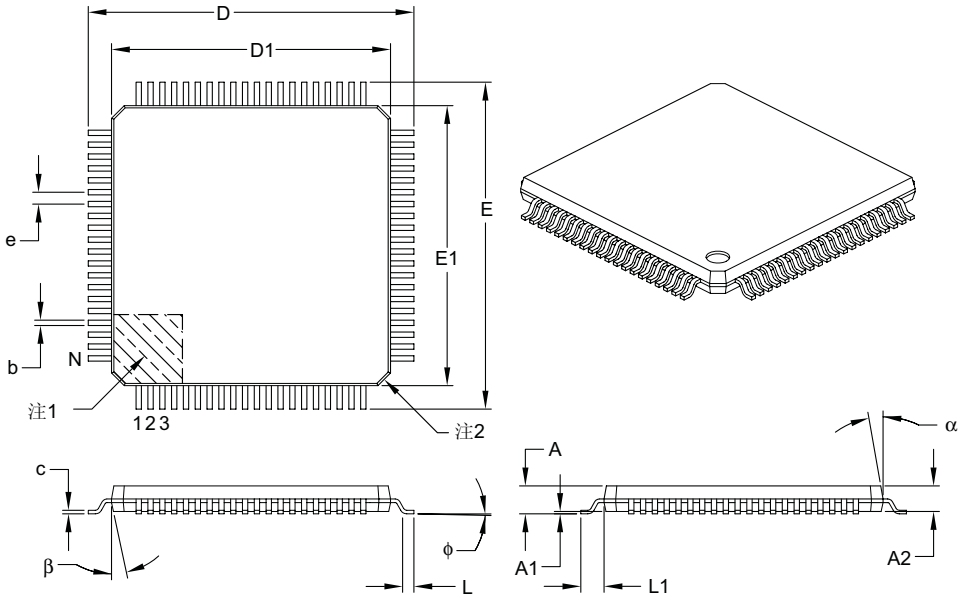
BSC：基本尺寸。显示的是没有公差的理论精确值。
REF：参考尺寸，通常无公差，仅供参考。

Microchip Technology 图号C04-085B

© 2007 Microchip Technology Inc.

80引脚塑封薄型正方扁平封装（**PT**）——**12x12x1 mm**主体，**2.00 mm**引脚投影长度 **[TQFP]**

注： 最新封装图请至http://www.microchip.com/packaging查看Microchip封装规范。



单位		毫米		
尺寸范围		最小	正常	最大
引脚数	N	80		
引脚间距	e	0.50 BSC		
总高度	A	–	–	1.20
塑模封装厚度	A2	0.95	1.00	1.05
悬空间隙	A1	0.05	–	0.15
底脚长度	L	0.45	0.60	0.75
引脚投影长度	L1	1.00 REF		
底脚倾斜角	ϕ	0°	3.5°	7°
总宽度	E	14.00 BSC		
总长度	D	14.00 BSC		
塑模封装宽度	E1	12.00 BSC		
塑模封装长度	D1	12.00 BSC		
引脚厚度	c	0.09	–	0.20
引脚宽度	b	0.17	0.22	0.27
塑模顶部锥度	α	11°	12°	13°
塑模底部锥度	β	11°	12°	13°

注：

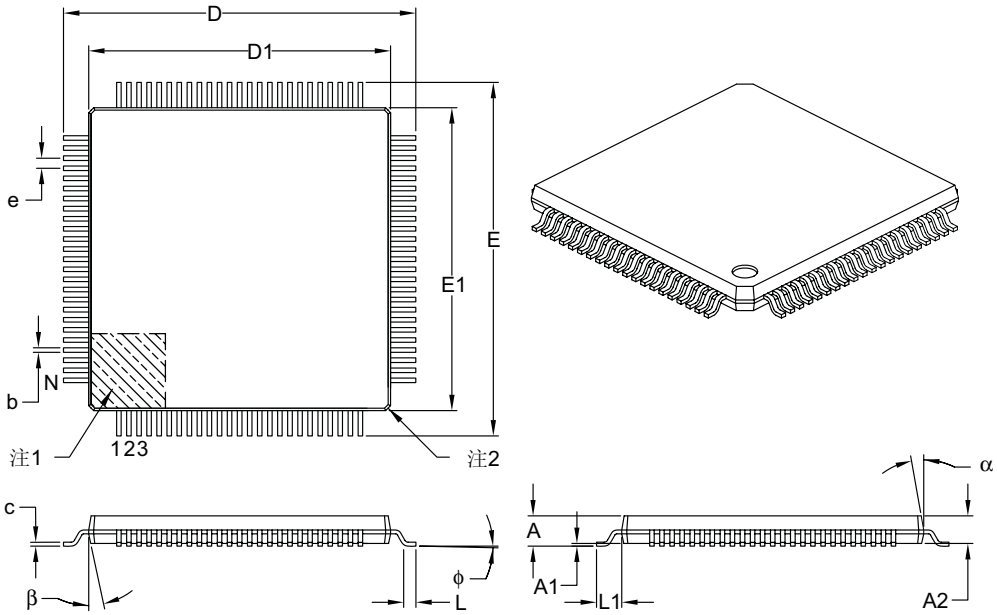
1. 引脚1的可见定位功能可能不同，但必须在阴影区域内。
2. 切角处的斜面是可选的；大小可能有所不同。
3. 尺寸D1和E1不包括塑模毛边或突起。 塑模每侧的毛边或突起不得超过0.25毫米。
4. 尺寸和公差请参见ASME Y14.5M。

BSC：基本尺寸。显示的是没有公差的理论精确值。
REF：参考尺寸，通常无公差，仅供参考。

Microchip Technology 图号C04-092B

100引脚塑封薄型正方扁平封装（**PT**）——**12x12x1 mm**主体，**2.00 mm**引脚投影长度 **[TQFP]**

注： 最新封装图请至http://www.microchip.com/packaging查看Microchip封装规范。



单位		毫米		
尺寸范围		最小	正常	最大
引脚数	N		100	
引脚间距	e		0.40 BSC	
总高度	A	–	–	1.20
塑模封装厚度	A2	0.95	1.00	1.05
悬空间隙	A1	0.05	–	0.15
底脚长度	L	0.45	0.60	0.75
引脚投影长度	L1		1.00 REF	
底脚倾斜角	ϕ	0°	3.5°	7°
总宽度	E		14.00 BSC	
总长度	D		14.00 BSC	
塑模封装宽度	E1		12.00 BSC	
塑模封装长度	D1		12.00 BSC	
引脚厚度	c	0.09	–	0.20
引脚宽度	b	0.13	0.18	0.23
塑模顶部锥度	α	11°	12°	13°
塑模底部锥度	β	11°	12°	13°

注：

1. 引脚1的可见定位功能可能不同，但必须在阴影区域内。
2. 切角处的斜面是可选的；大小可能有所不同。
3. 尺寸D1和E1不包括塑模毛边或突起。 塑模每侧的毛边或突起不得超过0.25毫米。
4. 尺寸和公差请参见ASME Y14.5M。

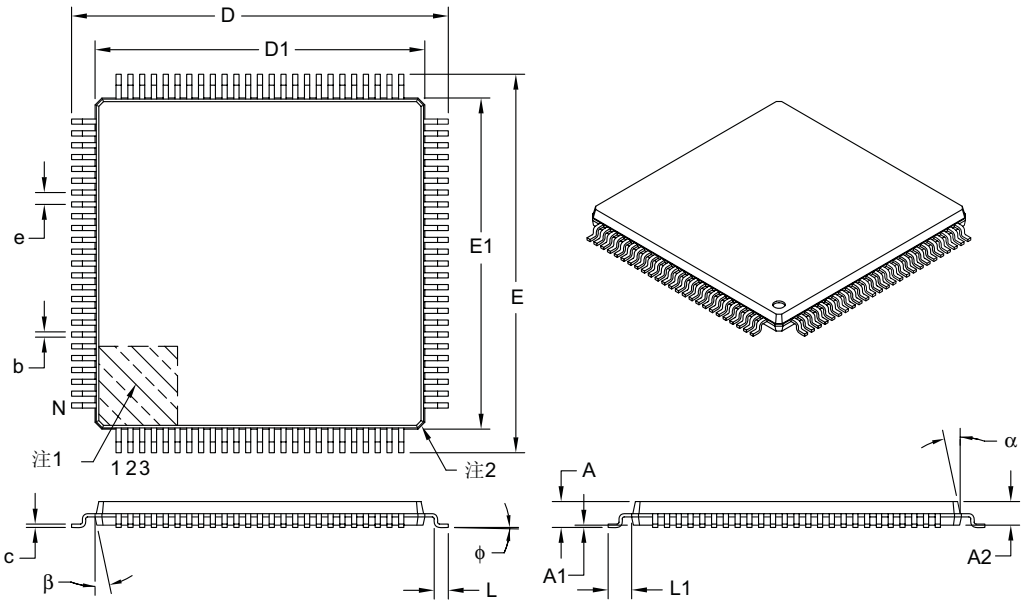
BSC：基本尺寸。显示的是没有公差的理论精确值。
REF：参考尺寸，通常无公差，仅供参考。

Microchip Technology 图号C04-100B

© 2007 Microchip Technology Inc.

100引脚塑封薄型正方扁平封装（**PF**）——**14x14x1 mm**主体，**2.00 mm**引脚投影长度 **[TQFP]**

注： 最新封装图请至http://www.microchip.com/packaging查看Microchip封装规范。



单位		毫米		
尺寸范围		最小	正常	最大
引脚数	N		100	
引脚间距	e		0.50 BSC	
总高度	A	–	–	1.20
塑模封装厚度	A2	0.95	1.00	1.05
悬空间隙	A1	0.05	–	0.15
底脚长度	L	0.45	0.60	0.75
引脚投影长度	L1		1.00 REF	
底脚倾斜角	ϕ	0°	3.5°	7°
总宽度	E		16.00 BSC	
总长度	D		16.00 BSC	
塑模封装宽度	E1		14.00 BSC	
塑模封装长度	D1		14.00 BSC	
引脚厚度	c	0.09	–	0.20
引脚宽度	b	0.17	0.22	0.27
塑模顶部锥度	α	11°	12°	13°
塑模底部锥度	β	11°	12°	13°

注：

1. 引脚1的可见定位功能可能不同，但必须在阴影区域内。
2. 切角处的斜面是可选的；大小可能有所不同。
3. 尺寸D1和E1不包括塑模毛边或突起。 塑模每侧的毛边或突起不得超过0.25毫米。
4. 尺寸和公差请参见ASME Y14.5M。

BSC：基本尺寸。显示的是没有公差的理论精确值。

REF：参考尺寸，通常无公差，仅供参考。

Microchip Technology 图号C04-110B

注：

© 2007 Microchip Technology Inc.

附录 A: “PS”（原型样片）器件和最终产品器件之间的差异

标注“PS”的 dsPIC33FJXXXGPX06/X08/X10 器件与最终产品器件（没有标注“PS”的器件）之间存在一些主要差异。本附录中列出了这些主要差异。此外，在 SFR 名称、位和复位状态方面还有一些小的差异，这在**第 3.0 节“存储器构成”**和对应的外设章节中进行说明。

A.1 器件名称

原型样片器件在其名称中含有后缀“PS”，正如器件封装上标识的那样。这将它们与工程样片器件（含有后缀“ES”）和最终产品器件（在器件封装标识上既没有“PS”后缀，也没有“ES”后缀）区分开来。

只有最终产品器件的一部分提供原型样片。请参见本数据手册中的器件表获取所有器件的列表。

A.2 RAM 容量

每种“PS”器件和对应的最终产品器件的 RAM 总容量（包括双口 DMA RAM 容量）是不同的。例如，最终产品器件具有 2 KB DMA RAM，而“PS”器件具有 1 KB DMA RAM。请参见本数据手册中的器件表获取每种 dsPIC33FJXXXGPX06/X08/X10 器件的存储器容量。

A.3 中断

最终产品器件与“PS”器件相比多了 4 个中断源（向量）。还有两个中断向量和与“PS”器件中对应中断稍有不同的事件相关联。更多详细信息，请参见**第 6.0 节“中断控制器”**。

A.4 DMA 增强功能

“PS”和最终产品器件都能执行直接存储器访问（DMA）数据传输。

除了“PS”器件中 DMA 控制器支持的所有特性外，最终产品器件中的DMA控制器还支持外设间接寻址模式。关于该特性的说明，请参见**第 7.0 节“直接存储器访问（DMA）”**。

A.5 振荡器操作

“PS”器件和最终产品器件的 PLL 后分频比和反馈倍频比位的默认值不同。关于寄存器定义和复位状态，请参见**第 8.0 节“振荡器配置”**。

A.6 CAN 和增强型 CAN

标注“PS”的 dsPIC33FJXXXGPX06/X08/X10 器件具有最多两个 CAN 模块。这些模块的功能和寄存器排列与 dsPIC30F 器件中的相同，在本数据手册的**第 18.0 节“增强型 CAN（ECAN™）模块”**中进行了说明。这些模块不提供 DMA 支持。

最终产品器件具有最多两个增强型 CAN（ECAN™ 技术）模块。这些模块具有比 CAN 模块明显更多的特性，主要体现在可用缓冲器、过滤器和屏蔽寄存器个数的增加，以及 DMA 支持。

A.7 ADC 差异

“PS”和最终产品器件都包含最多两个 ADC 模块。

“PS”器件具有一个 16 字深度 ADC 结果缓冲器。

最终产品器件具有以额外 DMA RAM 和外设间接寻址形式提供的增强型 DMA 支持。这实现了 16 字 ADC 缓冲器冗余。然而，缓冲器已由单 ADC 结果寄存器替换了。

A.8 器件封装

最终产品器件以下列 TQFP 封装形式提供：

- 64 引脚 TQFP 10x10x1 mm
- 80 引脚 TQFP 12x12x1 mm
- 100 引脚 TQFP 12x12x1 mm
- 100 引脚 TQFP 14x14x1 mm

“PS”器件以下列 TQFP 封装形式提供：

- 64 引脚 TQFP 10x10x1 mm
- 80 引脚 TQFP 12x12x1 mm
- 100 引脚 TQFP 14x14x1 mm

附录 B： 版本历史

版本 A（2007 年 5 月）

本文档的初始版本。

© 2007 Microchip Technology Inc.

索引

A

A/D 转换器 231
  DMA 231
    初始化 231
    主要特性 231
AC-Link 模式操作 224
  16 位模式 224
  20 位模式 225
ADC 模块
  ADC1 寄存器映射 45
  ADC2 寄存器映射 45

B

备用向量表 （AIVT） 79
变更通知客户服务 319

C

C 编译器
  MPLAB C18 262
  MPLAB C30 262
CPU
  控制寄存器 20
CPU 的特殊功能 245
CPU 时钟系统 136
  选择 136
操作码说明中使用的符号 254
程序存储器
  复位向量 31
  构成 31
  中断向量 31
程序地址空间 29
  表读指令
    TBLRDH 64
    TBLRDL 64
  存储器映射 30
  访问程序空间内数据的地址生成方式 63
  构成 62
  可视性操作 65
  使用表指令访问程序存储器中的数据 64
  使用程序空间可视性访问程序存储器中的数据 65
串行外设接口 （SPI） 161
存储器构成 29

D

DCI
  CPU 空闲模式期间的工作 224
  CPU 休眠模式期间的工作 224
  CSDO 模式位 224
  Slot 状态位 224
  采样时钟边沿控制位 222
  从器件帧同步操作 220
  发送 / 接收移位寄存器 217
  发送时隙使能位 222
  发送状态位 223
  工作 219
  缓冲器长度控制 223
  缓冲器控制 217
  缓冲器数据对齐 217
  缓冲器与数据帧对齐 223
  简介 217
  接收时隙使能位 222
  接收状态位 223
  模块使能 219
  数据对齐控制位 222
  数字环回模式 224
  同步数据传输 222
  位时钟发生器 221
  下溢模式控制位 224
  用于常见编解码器 CSCK 频率的器件频率 （表） 221
  帧同步发生器 219
  帧同步模式控制位 219
  帧同步时时隙使能位的操作 222
  中断 224
  主器件帧同步操作 219
  字长选择位 219
DCI I/O 引脚 217
  COFS 217
  CSCK 217
  CSDI 217
  CSDO 217
DCI 模块
  寄存器映射 54
DMAC 寄存器 126
  DMAxCNT 126
  DMAxCON 126
  DMAxPAD 126
  DMAxREQ 126
  DMAxSTA 126
  DMAxSTB 126
DMA 模块
  DMA 寄存器映射 46
DSP 引擎 23
  乘法器 25
代码保护 245, 251
代码示例
  PWRSAV 指令语法 143
  擦除程序存储器页 71
  端口写 / 读 146
  启动编程序列 72
  装载写缓冲器 72
电气特性 265
  AC 273
读者反馈 320

E

ECAN 模块
  ECAN1 寄存器映射 （C1CTRL1.WIN = 0 或 1） 48
  ECAN1 寄存器映射 （C1CTRL1.WIN = 0） 48
  ECAN1 寄存器映射 （C1CTRL1.WIN = 1） 49
  ECAN2 寄存器映射 （C2CTRL1.WIN = 0 或 1） 51
  ECAN2 寄存器映射 （C2CTRL1.WIN = 0） 51, 52
  报文发送 191
  报文接收 189
  波特率设置 192
  概述 187
  工作模式 189
  帧类型 187

F

FSCM
  晶振和 PLL 时钟源的延时 77
  器件复位 77
封装 305
  标识 305
  详细信息 306
复位 73
  时间 76

时钟源选择 76
特殊功能寄存器的复位状态 78
复位过程 79

G

公式
A/D 转换时钟周期 234
COFSG 周期 219
UART 波特率 （BRGH = 0） 180
UART 波特率 （BRGH = 1） 180
串行时钟速率 169
计算 PWM 周期 158
计算最大 PWM 分辨率 158
器件工作频率 136
器件工作频率和 SPI 时钟速度之间的关系 164
时钟发生的时间份额 193
位时钟频率 221
工作模式
初始化 189
环回 189
监听 189
监听所有报文 189
禁止 189
正常工作 189

H

红外支持
内置 IrDA 编码器和解码器 181
外部 IrDA， IrDA 时钟输出 181
汇编器
MPASM 汇编器 262

I

I/O 端口 145
并行 I/O （PIO） 145
写 / 读时序 146
I^2C
IPMI 支持 171
波特率发生器 169
从地址掩码 171
地址 171
工作模式 169
广播呼叫地址支持 171
寄存器 169
软件控制的时钟延长 （STREN = 1） 171
斜率控制 172
中断 169
主模式工作
多主器件通信、总线冲突和总线仲裁 172
时钟仲裁 172
I^2C 模块
I2C1 寄存器映射 43
I2C2 寄存器映射 43
I^2S 模式操作
数据对齐 225
I^2S 模式的操作 225
帧和数据字长度选择 225

J

寄存器
ADxCHS0 （ADCx 输入通道 0 选择） 241
ADxCHS123 （ADCx 输入通道 1、 2 和 3 选择） 240
ADxCON1 （ADCx 控制寄存器 1） 235
ADxCON2 （ADCx 控制寄存器 2） 237
ADxCON3 （ADCx 控制寄存器 3） 238
ADxCON4 （ADCx 控制寄存器 4） 239
ADxCSSH （ADCx 输入扫描选择寄存器的高位字） 242
ADxCSSL （ADCx 输入扫描选择寄存器的低位字） 242
ADxPCFGH （ADCx 端口配置寄存器的高位字） 243
ADxPCFGL （ADCx 端口配置寄存器的低位字） 243
CiBUFPNT1 （ECAN 过滤器 0-3 缓冲器指针） 204
CiBUFPNT2 （ECAN 过滤器 4-7 缓冲器指针） 205
CiBUFPNT3 （ECAN 过滤器 8-11 缓冲器指针） 205
CiBUFPNT4 （ECAN 过滤器 12-15 缓冲器指针） 206
CiCFG1 （ECAN 波特率配置寄存器 1） 202
CiCFG2 （ECAN 波特率配置寄存器 2） 203
CiCTRL1 （ECAN 控制寄存器 1） 194
CiCTRL2 （ECAN 控制寄存器 2） 195
CiEC （ECAN 发送 / 接收错误计数寄存器） 201
CiFCTRL （ECAN FIFO 控制寄存器） 197
CiFEN1 （ECAN 接收过滤器使能） 204
CiFIFO （ECAN FIFO 状态） 198
CiFMSKSEL1 （ECAN 过滤器 7-0 屏蔽选择） 208
CiINTE （ECAN 中断允许） 200
CiINTF （ECAN 中断标志） 199
CiRXFnEID （ECAN 接收过滤器 n 扩展标识符） 207
CiRXFnSID （ECAN 接收过滤器 n 标准标识符） 207
CiRXFUL1 （ECAN 接收缓冲器满 1） 210
CiRXFUL2 （ECAN 接收缓冲器满 2） 210
CiRXMnEID （ECAN 接收过滤器屏蔽器 n 扩展标识符） 209
CiRXMnSID （ECAN 接收过滤器屏蔽器 n 标准标识符） 209
CiRXOVF1 （ECAN 接收缓冲器溢出寄存器 1） 211
CiRXOVF2 （ECAN 接收缓冲器溢出寄存器 2） 211
CiTRBnDLC （ECAN 缓冲器 n 数据长度控制寄存器） 214
CiTRBnDm （ECAN 缓冲器 n 数据字段字节 m） 214
CiTRBnEID （ECAN 缓冲器 n 扩展标识符） 213
CiTRBnSID （ECAN 缓冲器 n 标准标识符） 213
CiTRBnSTAT （ECAN 接收缓冲器 n 状态） 215
CiTRmnCON （ECAN 发送 / 接收缓冲器 m 控制） 212
CiVEC （ECAN 中断编码） 196
CLKDIV （时钟分频比） 139
CORCON （内核控制） 22, 84
DCICON1 （DCI 控制 1） 226
DCICON2 （DCI 控制 2） 227
DCICON3 （DCI 控制 3） 228
DCISTAT （DCI 状态） 229
DMACS0 （DMA 控制器状态 0） 131
DMACS1 （DMA 控制器状态 1） 133
DMAxCNT （DMA 通道 x 传输计数） 130
DMAxCON （DMA 通道 x 控制） 127
DMAxPAD （DMA 通道 x 外设地址） 130
DMAxREQ （DMA 通道 x IRQ 选择） 128
DMAxSTA （DMA 通道 x RAM 起始地址 A） 129
DMAxSTB （DMA 通道 x RAM 起始地址 B） 129
DSADR （最近的 DMA RAM 地址） 134
I2CxCON （I2Cx 控制寄存器） 173
I2CxMSK （I2Cx 从模式地址掩码寄存器） 177
I2CxSTAT （I2Cx 状态寄存器） 175
ICxCON （输入捕捉 x 控制寄存器） 156
IEC0 （中断允许控制寄存器 0） 96
IEC1 （中断允许控制寄存器 1） 98
IEC2 （中断允许控制寄存器 2） 100
IEC3 （中断允许控制寄存器 3） 102
IEC4 （中断允许控制寄存器 4） 103
IFS0 （中断标志状态寄存器 0） 88

© 2007 Microchip Technology Inc.

IFS1 （中断标志状态寄存器 1）..... 90
IFS2 （中断标志状态寄存器 2）..... 92
IFS3 （中断标志状态寄存器 3）..... 94
IFS4 （中断标志状态寄存器 4）..... 95
INTCON1 （中断控制寄存器 1）..... 85
INTCON2 （中断控制寄存器 2）..... 87
INTTREG 中断控制和状态寄存器 122
IPC0 （中断优先级控制寄存器 0）..... 104
IPC10 （中断优先级控制寄存器 10）..... 114
IPC11 （中断优先级控制寄存器 11）..... 115
IPC12 （中断优先级控制寄存器 12）..... 116
IPC13 （中断优先级控制寄存器 13）..... 117
IPC14 （中断优先级控制寄存器 14）..... 118
IPC15 （中断优先级控制寄存器 15）..... 119
IPC16 （中断优先级控制寄存器 16）..... 120
IPC17 （中断优先级控制寄存器 17）..... 121
IPC1 （中断优先级控制寄存器 1）..... 105
IPC2 （中断优先级控制寄存器 2）..... 106
IPC3 （中断优先级控制寄存器 3）..... 107
IPC4 （中断优先级控制寄存器 4）..... 108
IPC5 （中断优先级控制寄存器 5）..... 109
IPC6 （中断优先级控制寄存器 6）..... 110
IPC7 （中断优先级控制寄存器 7）..... 111
IPC8 （中断优先级控制寄存器 8）..... 112
IPC9 （中断优先级控制寄存器 9）..... 113
NVMCOM （闪存存储器控制寄存器）..... 69, 70
OCxCON （输出比较 x 控制寄存器）..... 160
OSCCON （振荡器控制寄存器）..... 138
OSCTUN （FRC 振荡器调节寄存器）..... 141
PLLFBD （PLL 反馈倍频比寄存器）..... 140
RCON （复位控制寄存器）..... 74
RSCON （DCI 接收时隙控制寄存器）..... 230
SPIxCON1 （SPIx 控制寄存器 1）..... 166
SPIxCON2 （SPIx 控制寄存器 2）..... 167
SPIxSTAT （SPIx 状态和控制寄存器）..... 165
SR （CPU 状态寄存器）..... 20, 84
T1CON （Timer1 控制寄存器）..... 148
TSCON （DCI 发送时隙控制寄存器）..... 230
TxCON （T2CON、 T4CON、 T6CON 或 T8CON 控制寄存器）..... 152
TyCON （T3CON、 T5CON、 T7CON 或 T9CON 控制寄存器）..... 153
UxMODE （UARTx 模式寄存器）..... 182
UxSTA （UARTx 状态和控制寄存器）..... 184
JTAG 边界扫描接口 245
基于指令的节能模式 143
空闲 144
休眠 143
交流特性 273
负载条件 273
内部 RC 精度 275
节能特性 143
时钟频率和切换 143

K

开发支持 261
看门狗定时器 （WDT）..... 245, 250
编程注意事项 250
勘误表 12
客户通知服务 319
客户支持 319
框图
16 位 Timer1 模块 147
A/D 模块 232, 233
DCI 模块 218
dsPIC33F 14
dsPIC33F CPU 内核 18
DSP 引擎 24
ECAN 模块 188
PLL 137
SPI 162
Timer2 （16 位）..... 151
Timer2/3 （32 位）..... 150
UART 179
复位系统 73
共用端口结构 145
看门狗定时器 （WDT）..... 250
片上稳压器的连接 249
器件时钟 135, 137
输出比较 159
输入捕捉 155

L

灵活的配置 245
漏极开路配置 146

M

Microchip 因特网网站 319
MPLAB ASM30 汇编器、链接器和库管理器 262
MPLAB ICD 2 在线调试器 263
MPLAB ICE 2000 高性能通用在线仿真器 263
MPLAB PM3 器件编程器 263
MPLAB REAL ICE 在线仿真器系统 263
MPLAB 集成开发环境软件 261
MPLINK 目标链接器 /MPLIB 目标库管理器 262
脉宽调制模式 158
模寻址 58
W 地址寄存器选择 59
操作示例 59
起始地址和结束地址 59
应用 60

N

NVM 模块
寄存器映射 56
内部 RC 振荡器
与 WDT 一起使用 250

P

PICSTART 2 开发编程器 264
PICSTART Plus 开发编程器 264
PMD 模块
寄存器映射 56
POR 和长振荡器起振时间 77
PORTA
寄存器映射 54
PORTB
寄存器映射 54
PORTC
寄存器映射 55
PORTD
寄存器映射 55
PORTE
寄存器映射 55
PORTF
寄存器映射 55
PORTG
寄存器映射 56
PWM
占空比 158
周期 158
“PS”和最终产品器件之间的差异 311

配置寄存器映射 245
配置模拟端口引脚 146
配置位 245
    说明 （表） 246

R

软件堆栈指针，帧指针
    CALL 堆栈帧 57
软件模拟器 （MPLAB SIM） 262

S

SPI
    从，帧从连接 164
    从，帧主连接 164
    主 / 从连接 163
    主，帧主连接 163
SPI 模块
    SPI1 寄存器映射 44
    SPI2 寄存器映射 44
闪存程序存储器 67
    RTSP 工作原理 68
    编程算法 71
    表指令 67
    操作 68
    控制寄存器 68
设置产生单个输出脉冲 157
设置产生连续输出脉冲 157
时序规范
    10 位 A/D 转换要求 303
    12 位 A/D 转换要求 300
    CAN I/O 要求 295
    I2Cx 总线数据要求 （从模式） 291
    I2Cx 总线数据要求 （主模式） 289
    PLL 时钟 275
    SPIx 从模式 （CKE = 0）要求 285
    SPIx 从模式 （CKE = 1）要求 286
    SPIx 主模式 （CKE = 0）要求 283
    SPIx 主模式 （CKE = 1）要求 284
    Timer1 外部时钟要求 279
    Timer2、 Timer4、 Timer6 和 Timer8
        外部时钟要求 280
    Timer3、 Timer5、 Timer7 和 Timer9
        外部时钟要求 280
    复位、看门狗定时器、振荡器起振定时器、上电延时定时器和欠压复位要求 278
    简单输出比较 /PWM 模式要求 282
    输出比较要求 281
时序特性
    CLKO 和 I/O 276
时序图
    10 位 A/D 转换 （CHPS = 01， SIMSAM = 0， ASAM = 0， SSRC = 000） 301
    10 位 A/D 转换 （CHPS = 01， SIMSAM = 0， ASAM = 1， SSRC = 111， SAMC = 00001） 302
    12 位 A/D 转换 （ASAM = 0， SSRC = 000） 299
    CAN I/O 295
    DCI AC-Link 模式 294
    DCI 多通道， I2S 模式 292
    ECAN 位 192
    I2Cx 总线启动位 / 停止位 （从模式） 290
    I2Cx 总线启动位 / 停止位 （主模式） 288
    I2Cx 总线数据 （从模式） 290
    I2Cx 总线数据 （主模式） 288
    I2S 接口帧同步 220
    SPIx 从模式 （CKE = 0） 285
    SPIx 从模式 （CKE = 1） 286
    SPIx 主模式 （CKE = 0） 283
    SPIx 主模式 （CKE = 1） 284
    Timer1, 2, 3, 4, 5, 6, 7, 8 和 9 外部时钟 279
    复位、看门狗定时器、振荡器起振定时器和上电延时定时器 277
    输出比较 /PWM 282
    输出比较 （OCx） 281
    输入捕捉 （CAPx） 281
    外部时钟 274
    帧同步， AC-Link 帧起始 220
    帧同步，多通道模式 220
时序要求
    CLKO 和 I/O 276
    DCI AC-Link 模式 294, 296
    DCI 多通道， I2S 模式 293, 296
    输入捕捉 281
    外部时钟 274
时钟切换 142
    使能 142
    序列 142
输出比较 157
    寄存器 160
数据地址空间 32
    Near 数据空间 32
    带有 16 KB RAM 的 dsPIC33F 器件的存储器映射 34
    带有 30 KB RAM 的 dsPIC33F 器件的存储器映射 35
    带有 8 KB RAM 的 dsPIC33F 器件的存储器映射 33
    对齐 32
    宽度 32
    软件堆栈 57
数据累加器和加法器 / 减法器 25
    回写 26
    舍入逻辑 26
    数据空间写饱和 27
    溢出和饱和 25
数据转换器接口 （DCI）模块 217
输入捕捉
    寄存器 156
输入状态变化通知模块 146
算术逻辑单元 （ALU） 23

T

Timer1 147
Timer2/3、 Timer4/5、 Timer6/7 和 Timer8/9 149
桶形移位寄存器 27

U

UART
    8 位或 9 位数据模式下的接收 181
    8 位数据模式下的发送 181
    9 位数据模式下的发送 181
    波特率
        发生器 （BRG） 180
    间隔和同步发送序列 181
    使用 UxCTS 和 UxRTS 引脚的流控制 181
UART 模块
    UART1 寄存器映射 44
    UART2 寄存器映射 44

W

WWW 地址 319
WWW，在线支持 12
外设模块禁止 （PMD） 144
位反转寻址 60
    示例 61
    实现 60

© 2007 Microchip Technology Inc.

序列表 （16 项） 61
温度和电压规范
AC 273
稳压器 （片上） 249

X

系统控制
寄存器映射 56

Y

引脚说明 （表） 15
因特网地址 319

Z

在节能指令执行期间的中断 144
在线串行编程 （ICSP） 245, 251
在线调试器 251
在线仿真 245
增强型 CAN 模块 187
指令集
概述 256
汇总 253
指令寻址模式 57
MAC 指令 58
MCU 指令 57
其他指令 58
文件寄存器指令 57
支持的基本模式 58
传送指令和累加器指令 58
直流特性 266
I/O 引脚输出规范 271
I/O 引脚输入规范 270
程序存储器 272
打盹电流 （IDOZE） 269
掉电电流 （IPD） 268
工作电流 （IDD） 267
空闲电流 （IIDLE） 268
温度和电压规范 266
中断控制和状态寄存器 83
IECx 83
IFSx 83
INTCON1 83
INTCON2 83
IPCx 83
中断设置过程 123
初始化 123
陷阱服务程序 123
中断服务程序 123
中断禁止 123
中断向量表 （IVT） 79
自动时钟延长 171
发送模式 171
接收模式 171

注：

© 2007 Microchip Technology Inc.

MICROCHIP 网站

Microchip 网站（www.microchip.com）为客户提供在线支持。客户可通过该网站方便地获取文件和信息。只要使用常用的因特网浏览器即可访问。网站提供以下信息：

- **产品支持**——数据手册和勘误表、应用笔记和示例程序、设计资源、用户指南以及硬件支持文档、最新的软件版本以及存档软件
- **一般技术支持**——常见问题（FAQ）、技术支持请求、在线讨论组以及 Microchip 顾问计划成员名单
- **Microchip 业务**——产品选型和订购指南、最新 Microchip 新闻稿、研讨会和活动安排、 Microchip 销售办事处、代理商以及工厂代表列表

变更通知客户服务

Microchip 的客户通知服务有助于客户了解 Microchip 产品的最新信息。注册客户可在他们感兴趣的某个产品系列或开发工具发生变更、更新、发布新版本或勘误表时，收到电子邮件通知。

欲注册，请登录 Microchip 网站 www.microchip.com，点击“变更通知客户（Customer Change Notification）”服务并按照注册说明完成注册。

客户支持

Microchip 产品的用户可通过以下渠道获得帮助：

- 代理商或代表
- 当地销售办事处
- 应用工程师（FAE）
- 技术支持

客户应联系其代理商、代表或应用工程师（FAE）寻求支持。当地销售办事处也可为客户提供帮助。本文档后附有销售办事处的联系方式。

也可通过http://support.microchip.com获得网上技术支持。

读者反馈表

我们努力为您提供最佳文档，以确保您能够成功使用 Microchip 产品。如果您对文档的组织、条理性、主题及其他有助于提高文档质量的方面有任何意见或建议，请填写本反馈表并传真给我公司 TRC 经理，传真号码为 86-21-5407-5066。

请填写以下信息，并从下面各方面提出您对本文档的意见。

致： TRC 经理　　总页数 _____

关于： 读者反馈

发自： 姓名 _____

公司 _____

地址 _____

国家 / 省份 / 城市 / 邮编 _____

电话：(_____) _____ - _____　　传真：(_____) _____ - _____

应用（选填）：

您希望收到回复吗？是 ____ 否____

器件： dsPIC33FJXXXGPX06/X08/　　文献编号： DS70286A_CN

问题：

1. 本文档中哪些部分最有特色？

2. 本文档是否满足了您的软硬件开发要求？如何满足的？

3. 您认为本文档的组织结构便于理解吗？如果不便于理解，那么问题何在？

4. 您认为本文档应该添加哪些内容以改善其结构和主题？

5. 您认为本文档中可以删减哪些内容，而又不会影响整体使用效果？

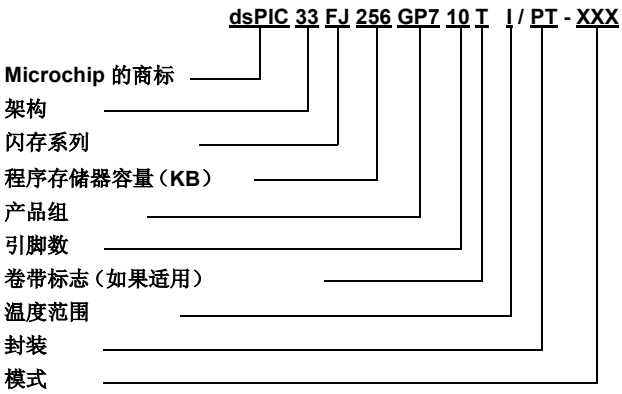
6. 本文档中是否存在错误或误导信息？如果存在，请指出是什么信息及其具体页数。

7. 您认为本文档还有哪些方面有待改进？

© 2007 Microchip Technology Inc.

产品标识体系

欲订货或获取价格、交货等信息，请与我公司生产厂或各销售办事处联系。



架构：	33	=	16 位数字信号控制器
闪存系列：	FJ	=	闪存程序存储器， 3.3V
产品组：	GP2	=	通用系列
	GP3	=	通用系列
	GP5	=	通用系列
	GP7	=	通用系列
引脚数：	06	=	64 引脚
	08	=	80 引脚
	10	=	100 引脚
温度范围：	I	=	-40°C 至 +85°C （工业级）
封装：	PT	=	10x10 或 12x12 mm TQFP （薄型正方扁平封装）
	PF	=	14x14 mm TQFP （薄型正方扁平封装）
模式：	三位 QTP、 SQTP、代码或特殊要求（其他情况空白）		
	ES	=	工程样片

示例：

a) dsPIC33FJ256GP710I/PT：
通用 dsPIC33 系列器件， 64 KB 程序存储器，
100 引脚，工业级温度， TQFP 封装。

全球销售及服务网点

深圳市英锐恩科技有限公司（www.enroo.com）
(Microchip Authorized Design Partner)指定授权代理
Add： Room 1203-1205 Top office,Glittery City,No. 3027,
Shennan Road Central,Futian , Shenzhen City
电话(tel) ：86-755-88845951,82543411
传真(fax) ：86-755-82543511
Web: Http://www.Enroo.com ,"o"为字母.
E-mail： enroo@enroo.com jason.ma@139.com
联系人：马先生,王小姐
公司在线咨询：QQ:27781279 MSN:picmcu@hotmail.com
7x24小时在线产品咨询:13510398583 13798484366

© 2007 Microchip Technology Inc.